

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ตัวกรองความถี่เชิงเลขโดยใช้ เอฟพีจีเอ

Digital filter using FPGA



เลขหมู่.....
เลขทะเบียน..... 72090
วัน,เดือน,ปี..... - 8 ส.ย. 2550

b. 117 b3310
i.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาอิเล็กทรอนิกส์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวกรองความถี่เชิงเลขโดยใช้ เอฟพีจีเอ

Digital filter using FPGA



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2549

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ตัวกรองความถี่เชิงเลข โดยใช้เอฟพีจีเอ

Digital filter using FPGA

ผู้จัดทำ

1. นายอรรคพล ธนารักษ์ 46010934

2. นายอรรคภัทร์ ลิขิตสังขกุล 46010935

ลงชื่อ..... อาจารย์ที่ปรึกษา

(ศ.ดร. วัลลภ สุระกำพลร)

วันที่ 4 / 4 / 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวกรองความถี่เชิงเลขโดยใช้ เอฟพีจีเอ

นาย อรรคพล ธนารักษ์ รหัส 46010934
นาย อรรคภัทร์ ลิขิตสังจากุล รหัส 46010935
ศ.ดร. วัลลภ สุระกำพลธร อาจารย์ที่ปรึกษา
ปีการศึกษา 2549

บทคัดย่อ

รายงานฉบับนี้ศึกษาการประยุกต์ใช้ความรู้เรื่องการประมวลผลสัญญาณเชิงเลขในการสร้างวงจรอีควอไลเซอร์ โดยมีหลักการทำงาน คือ รับสัญญาณในช่วงความถี่เสียงตั้งแต่ 20 เฮิร์ตซ์ ถึง 20 กิโลเฮิร์ตซ์ เข้าสู่วงจรแปลงสัญญาณเชิงอุปมานเป็นสัญญาณเชิงเลข ผ่านวงจรกรองสัญญาณเชิงเลขแบบความถี่ผ่านแถบ จากนั้นจึงส่งผ่านตัวขยายสัญญาณที่สามารถเปลี่ยนค่าได้ตั้งแต่ -20 เดซิเบล ถึง +20 เดซิเบล แล้วจึงนำสัญญาณทั้งหมดมารวมกันก่อนแปลงกลับจากสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมานเพื่อใช้งานต่อไป โดยได้ทำการศึกษาและออกแบบระบบด้วย ซอร์ฟแวร์แมทแลบ แล้วจึงทำการสร้างเฉพาะวงจรกรองความถี่ซึ่งเป็นสิ่งสำคัญที่สุดของอีควอไลเซอร์บนไอซีเอฟพีจีเอ

Digital filter using FPGA

Mr. Arkkapol Dhanaraks ID.46010934

Mr. Arkapat Likitsadjakul ID.46010935

Prof.Dr. Wanlop Surakampomthorn Adviser

Educational Year 2006

Abstract

This report studies on the applying of digital signal processing for designing a fully programmable digital Equalizer. The system operates by receiving audio frequency from 20 Hertz to 20 Kilohertz and converting to digital domain by an analog to digital converter. The converted signal will pass through digital bank filter, after that the signal in each band will be amplified separately, which is able to vary an amplitude from -20 dB to +20 dB. The amplified signal will be summed together before sent to processing in the next stage or converted back to analog with a digital to analog converter. System in digital signal processing are studied and designed in Matlab and worked on Field Programable Gate Array (FPGA) IC.

กิตติกรรมประกาศ

โครงการนี้จะไม่มีทางสำเร็จลุล่วงไปได้หากปราศจาก พ่อ แม่ ผู้สนับสนุนด้านการศึกษา อย่างดีเสมอมา อาจารย์วัลลภ สุระกำพลธร ผู้ให้ความช่วยเหลือในด้านต่างๆ นายอมร จิรเสรีอมร กุลผู้ให้คำปรึกษา เสนอแนวทางในการแก้ปัญหาและคอยดูแลการทำโครงการนี้ให้ดี สุดท้ายนี้ ขอขอบคุณ นายอรรถพล วีระนพนันท์ และเพื่อนๆ ที่ให้ข้อมูลที่มีประโยชน์กับโครงการนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.2.5 FPGA	36
บทที่ 3 การทดลอง (ทฤษฎี)	37
3.1 การทดลองที่ 1	37
3.2 การทดลองที่ 2	41
3.3 การทดลองที่ 3	42
บทที่ 4 ผลการทดลอง (ทฤษฎี)	44
4.1 ผลการทดลองอิควอไลเซอร์ที่มีลำดับของตัวกรองต่ำ (53) กับสัญญาณชานน์	44
4.2 ผลการทดลองอิควอไลเซอร์ที่มีลำดับของตัวกรองสูง (2000) กับสัญญาณชานน์	51
บทที่ 5 สรุปผลการทดลอง (ทฤษฎี)	57
5.1 สรุปผลการทดลอง	57
5.2 ปัญหาที่พบ	57
5.3 แนวทางการแก้ปัญหา	57
บทที่ 6 สถาปัตยกรรมและการออกแบบบน FPGA	58
6.1 เอฟจีพีเอ (FPGA: Field Programmable Gate Array)	59
6.1.1 เปรียบเทียบกับ ไมโครคอนโทรลเลอร์	59
6.1.2 จุดเด่นของ FPGA	60
6.1.3 FPGA ในท้องตลาดและการเลือกใช้	60
6.2 รายละเอียดสถาปัตยกรรมของ FPGA ตระกูล Spartan II	60
6.2.1 บล็อกอินพุท/เอาต์พุท (IOBs)	61
6.2.2 ลอจิกเซลล์ (Logic cells)	62
6.3 การออกแบบวงจรดิจิทัลในปัจจุบัน	64
6.3.1 การออกแบบโดยใช้การวาดผังวงจร (Schematic design)	64
6.3.2 การออกแบบโดยใช้ภาษาระดับสูง (HDL)	65
6.4 การเขียนภาษา HDL	66
6.4.1 ลักษณะเฉพาะในการออกแบบ (Design Specification)	66

สารบัญ (ต่อ)

	หน้า
6.4.2 การจำลองการทำงานฟังก์ชัน (Function simulation)	67
6.4.3 การสังเคราะห์วงจร (Synthesis)	67
6.4.4 Timing simulation	68
6.4.5 Place & Route	68
6.5 การออกแบบจากบนลงล่างและการออกแบบจากล่างขึ้นบน	68
6.6 RTL (Register Transfer Level)	69
บทที่ 7 การออกแบบชิ้นงาน	71
7.1. วงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล	71
7.2 ตัวกรองสัญญาณ	72
7.2.1 ส่วนแปลงเลขฐานสองแบบ วันคอมพลิเมนต์เป็น ทวคอมพลิเมนต์	72
7.2.2 ตัวลำดับการจ่ายสัญญาณ (Splitter)	73
7.2.3 ตัวคูณ (Multiplier)	74
7.2.3.1 ตัวแยกบิต	75
7.2.3.2 ตารางเปิดดู (Look up Table)	76
7.2.3.3 ตัวเลื่อนสัญญาณ (Shift)	77
7.2.3.4 ตัวรวมสัญญาณ (Sum)	78
7.2.4 แปลงเลขฐานสองแบบ ทวคอมพลิเมนต์เป็นวันคอมพลิเมนต์	79
7.2.5 ตัวรวมสัญญาณ (Sum)	80
7.3 วงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก	84
7.4 เส้นทางของข้อมูลทางดิจิทัล	85
บทที่ 8 ผลการทดลอง	86
8.1 ตอนที่ 1 สัญญาณที่ผ่านตัวแปลงสัญญาณอนาล็อกเป็นดิจิทัลและ ตัวแปลงสัญญาณดิจิทัลเป็นอนาล็อก	86
8.2 ตอนที่ 2 สัญญาณที่ผ่านตัวกรองความถี่แบบแถบผ่าน	91
8.3 ตอนที่ 3 กราฟแสดงผลการตอบสนองทางความถี่	99

สารบัญ (ต่อ)

	หน้า
บทที่ 9 บทสรุป	101
9.1 บทสรุป	101
9.2 ปัญหาในโครงการ	101
9.3 แนวทางการแก้ปัญหา	101
บรรณานุกรม	



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 2.1 บล็อกไดอะแกรมของอีควอไลเซอร์	3
รูปที่ 2.2 บล็อกไดอะแกรมของระบบประมวลผลสัญญาณดิจิทัล	5
รูปที่ 2.3 บล็อกไดอะแกรมของระบบประมวลผลสัญญาณดิจิทัลที่สมบูรณ์	5
รูปที่ 2.4 สัญญาณต่อเนื่องและสัญญาณไม่ต่อเนื่อง	6
รูปที่ 2.5 ฟังก์ชันอิมพัลส์หนึ่งหน่วย	7
รูปที่ 2.6 ฟังก์ชันขั้นหนึ่งหน่วย	8
รูปที่ 2.7 ฟังก์ชันเอกซ์โพเนนเชียล	9
รูปที่ 2.8 ฟังก์ชันซายน์	10
รูปที่ 2.9 สัญญาณแบบมีคาบ	11
รูปที่ 2.10 สัญญาณสมมาตร	11
รูปที่ 2.11 สัญญาณไม่สมมาตร	12
รูปที่ 2.12 บล็อกไดอะแกรมของระบบแบบไม่ต่อเนื่อง	14
รูปที่ 2.13 บล็อกไดอะแกรมการบวกในระบบแบบไม่ต่อเนื่อง	14
รูปที่ 2.14 บล็อกไดอะแกรมตัวคูณคงที่ในระบบแบบไม่ต่อเนื่อง	15
รูปที่ 2.15 บล็อกไดอะแกรมตัวคูณสัญญาณในระบบแบบไม่ต่อเนื่อง	15
รูปที่ 2.16 บล็อกไดอะแกรมตัวห่น่วงในระบบไม่ต่อเนื่อง	15
รูปที่ 2.17 บล็อกไดอะแกรมตัวล้าหน้าในระบบไม่ต่อเนื่อง	15
รูปที่ 2.18 ระบบไม่ต่อเนื่องที่แปรตามเวลาและไม่แปรตามเวลา	17
รูปที่ 2.19 แสดงระบบ τ เป็นระบบที่เชิงเส้น	18
รูปที่ 2.20 การเชื่อมต่อระบบไม่ต่อเนื่องแบบอนุกรม	19
รูปที่ 2.21 การเชื่อมต่อระบบไม่ต่อเนื่องแบบขนาน	19
รูปที่ 2.22 การแทนสัญญาณแบบไม่ต่อเนื่องในรูป Impulse	21
รูปที่ 2.23 คุณสมบัติการสลับที่	22
รูปที่ 2.24 คุณสมบัติการจับกลุ่ม	23
รูปที่ 2.25 คุณสมบัติการกระจาย	23
รูปที่ 2.26 โครงสร้างของตัวกรองเชิงเลข FIR	27
รูปที่ 2.27 แสดงผลตอบสนองอิมพัลส์ของเฟสเชิงเส้นของตัวกรอง 4 ชนิด	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 2.28 แสดงผลตอบสนองความถี่และผลตอบสนองอิมพัลส์	31
รูปที่ 2.29 แสดงผลการตอบสนองความถี่	33
รูปที่ 2.30 แสดงผลการตอบสนองความถี่ด้วยฟังก์ชันวิน โคว์	34
รูปที่ 2.31 แสดงกราฟผลตอบสนองความถี่ของตัวกรองแบบ Low-pass 2 ความถี่ และกราฟผลตอบสนองความถี่ของตัวกรองแบบ Band-pass	35
รูปที่ 3.1 โมเดลของอีควอไลเซอร์ที่ใช้อินพุทเป็นสัญญาณชายน์	37
รูปที่ 3.2 โมเดลสร้างสัญญาณชายน์	38
รูปที่ 3.3 ตัวกรองความถี่ทั้ง 10 ของสัญญาณ	38
รูปที่ 3.4 ส่วนขยายสัญญาณ	39
รูปที่ 3.5 บล็อกตัวกรองสัญญาณที่มีตัวกรองลำดับต่ำ (53)	40
รูปที่ 3.6 บล็อกตัวกรองสัญญาณที่มีตัวกรองลำดับสูง (2000)	41
รูปที่ 3.7 โมเดลของอีควอไลเซอร์ที่ใช้อินพุทเป็นไฟล์เสียงจริง	42
รูปที่ 4.1 สัญญาณชายน์อินพุท	44
รูปที่ 4.2 สัญญาณรวมชายน์อินพุท	44
รูปที่ 4.3 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 33Hz	45
รูปที่ 4.4 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 67Hz	45
รูปที่ 4.5 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 134Hz	46
รูปที่ 4.6 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 267Hz	46
รูปที่ 4.7 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 532Hz	47
รูปที่ 4.8 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 1055Hz	47
รูปที่ 4.9 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 2110Hz	48
รูปที่ 4.10 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 4220Hz	48
รูปที่ 4.11 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 8410Hz	49
รูปที่ 4.12 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 16800Hz	49
รูปที่ 4.13 สัญญาณรวมของสัญญาณเอาต์พุท	50
รูปที่ 4.14 สัญญาณรวมชายน์อินพุท	51
รูปที่ 4.15 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 33Hz	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 4.16 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 67Hz	52
รูปที่ 4.17 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 134Hz	52
รูปที่ 4.18 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 267Hz	53
รูปที่ 4.19 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 532Hz	53
รูปที่ 4.20 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 1055Hz	54
รูปที่ 4.21 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 2110Hz	54
รูปที่ 4.22 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 4220Hz	55
รูปที่ 4.23 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 8410Hz	55
รูปที่ 4.24 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 16800Hz	56
รูปที่ 4.25 สัญญาณรวมของสัญญาณเอาท์พุท	56
รูปที่ 6.1 การแบ่งกลุ่มของวงจรรวม ASIC	58
รูปที่ 6.2 บล็อกไดอะแกรมของ โครงสร้าง FPGA ตระกูล Spartan II	61
รูปที่ 6.3 บล็อกอินพุท/เอาท์พุทของ โครงสร้าง FPGA ตระกูล Spartan II	62
รูปที่ 6.4 บล็อก CLB slice ของ FPGA ตระกูล Spartan II (สมมาตรในแต่ละ CLB)	63
รูปที่ 6.5 ขั้นตอนการออกแบบวงจรด้วยภาษา HDL	66
รูปที่ 6.6 กระบวนการออกแบบ (Design Methodology)	68
รูปที่ 6.7 ตัวอย่าง Register Transfer Level บล็อก ไดอะแกรม	69
รูปที่ 7.1 บล็อกไดอะแกรมของอีควอไลเซอร์	71
รูปที่ 7.2 รูปวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณเชิงเลข	71
รูปที่ 7.3 บล็อกไดอะแกรมของตัวกรองสัญญาณ	72
รูปที่ 7.4 บล็อกไดอะแกรมของตัวคูณสัญญาณ	75
รูปที่ 7.5 โปรแกรมของตัวคูณสัญญาณ	82
รูปที่ 7.6 โปรแกรมของตัวกรองสัญญาณ	82
รูปที่ 7.7 ค่าผลการตอบสนองทางความถี่ที่ได้จากการออกแบบตัวกรองสัญญาณ ด้วยโปรแกรม Matlab	84
รูปที่ 7.8 รูปวงจรแปลงสัญญาณจากดิจิทัลเป็นอนาล็อก	84
รูปที่ 7.9 การแปลงข้อมูลแบบวันคอมพลิเมนต์และทวคอมพลิเมนต์	85

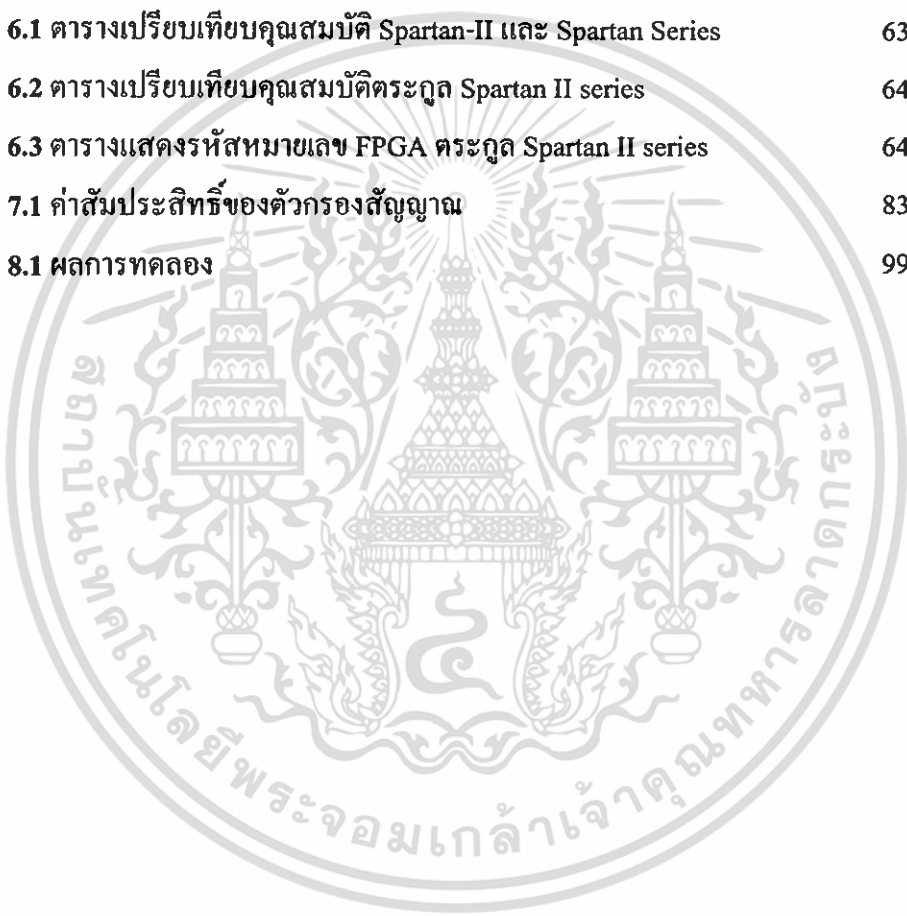
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 8.1 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 20 Hz	86
รูปที่ 8.2 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 100 Hz	87
รูปที่ 8.3 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 500 Hz	87
รูปที่ 8.4 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 1 kHz	88
รูปที่ 8.5 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 2 kHz	88
รูปที่ 8.6 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 5 kHz	89
รูปที่ 8.7 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 10 kHz	89
รูปที่ 8.8 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 15 kHz	90
รูปที่ 8.9 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 20 kHz	90
รูปที่ 8.10 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 20 Hz	91
รูปที่ 8.11 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 50 Hz	91
รูปที่ 8.12 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 100 Hz	92
รูปที่ 8.13 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 200 Hz	92
รูปที่ 8.14 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 500 Hz	93
รูปที่ 8.15 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 800 Hz	93
รูปที่ 8.16 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 1 kHz	94
รูปที่ 8.17 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 2 kHz	94
รูปที่ 8.18 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 5 kHz	95
รูปที่ 8.19 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 8 kHz	95
รูปที่ 8.20 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 10 kHz	96
รูปที่ 8.21 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 12 kHz	96
รูปที่ 8.22 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 15 kHz	97
รูปที่ 8.23 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 18 kHz	97
รูปที่ 8.24 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 20 kHz	98
รูปที่ 8.25 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 22 kHz	98
รูปที่ 8.26 ผลการตอบสนองทางความถี่	100

สารบัญตาราง

	หน้า
ตารางที่ 2.1 แสดงช่วงความถี่ออกเคฟตามมาตรฐานของ ISO	4
ตารางที่ 2.2 z-transform ของสัญญาณแบบไม่ต่อเนื่อง	25
ตารางที่ 2.3 เปรียบเทียบตัวกรองเชิงเลข	26
ตารางที่ 2.4 เปรียบเทียบฟังก์ชันหน้าต่าง	35
ตารางที่ 6.1 ตารางเปรียบเทียบคุณสมบัติ Spartan-II และ Spartan Series	63
ตารางที่ 6.2 ตารางเปรียบเทียบคุณสมบัติตระกูล Spartan II series	64
ตารางที่ 6.3 ตารางแสดงรหัสหมายเลข FPGA ตระกูล Spartan II series	64
ตารางที่ 7.1 ค่าสัมประสิทธิ์ของตัวกรองสัญญาณ	83
ตารางที่ 8.1 ผลการทดลอง	99



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

อีควอไลเซอร์ (Equalizer) เป็นอุปกรณ์ที่ได้รับความนิยมกันอย่างแพร่หลายทั้งในด้านงานบันเทิงและในงานด้านอุตสาหกรรม คุณสมบัติของอีควอไลเซอร์ คือ สามารถชดเชยการสูญเสียสัญญาณในช่วงความถี่ใดช่วงความถี่หนึ่ง หรือ สามารถปรับระดับสัญญาณในช่วงความถี่ที่เราต้องการได้ แต่อีควอไลเซอร์ไม่ได้ให้ความสำคัญกับสัญญาณรบกวนที่เกิดขึ้น ทำให้สัญญาณที่ได้มีความผิดพลาดสูง อีควอไลเซอร์แบบเดิมที่ใช้การประมวลสัญญาณแบบอนาล็อกจะต้องใช้อุปกรณ์จำนวนมาก เช่น ตัวขยายสัญญาณ หรือ DAC (Digital to Analog Converter) ต้องใช้จำนวนเท่ากับจำนวนช่วงสัญญาณ ทำให้เกิดการสิ้นเปลือง

โครงการนี้จะเป็นการศึกษาและทดลองสร้างอีควอไลเซอร์โดยใช้การประมวลผลเชิงเลขทั้งหมด โดยจะแบ่งช่วงความถี่ออกเป็น 10 ช่วงสัญญาณและแต่ละช่วงสัญญาณสามารถปรับอัตราขยายหรือลดทอนได้ 20 dB ซึ่งจะทำการสร้างลงบนชิพ FPGA (Field Programmable Gate Array) ซึ่งจะมีข้อดีคือ

- แต่ละช่วงสัญญาณสามารถปรับเปลี่ยน ช่วงความถี่และขนาดของสัญญาณเป็นอิสระจากกัน จึงสามารถนำไปประยุกต์ใช้งานได้หลากหลาย โดยเฉพาะกับงานที่ให้ความสำคัญกับบางช่วงสัญญาณมากเป็นพิเศษ เช่น ระบบเครื่องช่วยฟังสำหรับผู้มีปัญหาด้านการได้ยิน เป็นต้น
- สามารถบรรจุทั้งระบบลงในชิพ FPGA เพียงตัวเดียวได้ ทำให้ช่วยลดขนาดและจำนวนอุปกรณ์ลงได้มาก

1.2 วัตถุประสงค์ของการศึกษา

โครงการนี้มีวัตถุประสงค์เพื่อสร้างอีควอไลเซอร์ซึ่งใช้วงจรกรองความถี่แถบผ่านแบบ FIR หลายช่วงความถี่ที่สามารถปรับอัตราขยายในแต่ละช่วงความถี่แยกกันได้ โดยได้ทำการศึกษาและออกแบบระบบด้วยซอฟต์แวร์ MATLAB แล้วจึงทำการแปลงรหัสเพื่อสร้างเป็นวงจรจริงบนชิพ FPGA ต่อไป

1.3 ขอบเขตของโครงการ

- 1.3.1 ออกแบบตัวกรองเชิงเลขแบบความถี่แถบผ่านแบบ FIR โดยใช้ FPGA
- 1.3.2 ทดลองตัวกรองความถี่ด้วยความถี่ต่างๆ

1.4 ขั้นตอนการศึกษา

ขั้นตอนการศึกษามีดังนี้

- 1.4.1 ศึกษาหลักการของอีควอไลเซอร์
- 1.4.2 ศึกษาหลักการของตัวกรองเชิงเลข FIR
- 1.4.3 สร้างอีควอไลเซอร์โดยใช้โปรแกรม MATLAB และ ทดลองกับสัญญาณขาบ่นและเสียงพูดจริง
- 1.4.4 ศึกษาชิพ FPGA
- 1.4.5 สร้างวงจรกรองความถี่โดยใช้ FPGA และทดลองใช้จริง
- 1.4.6 เปรียบเทียบผลการทดลองที่ได้จาก MATLAB และการสร้างจริง แล้วจึงทำการวิเคราะห์สุดท้ายและสรุปผลการทดลอง

บทที่ 2

ทฤษฎีและหลักการ

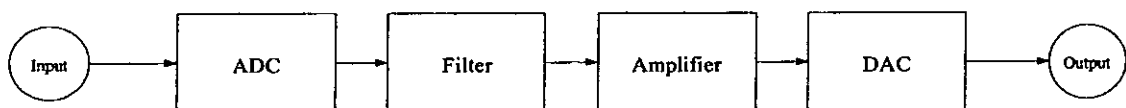
2.1 อีควอไลเซอร์

อีควอไลเซอร์ คือ อุปกรณ์ที่ใช้เปลี่ยนผลตอบสนองเชิงความถี่ของสัญญาณที่ผ่านเข้ามา เพื่อชดเชยการสูญเสียพลังงานในบางช่วงความถี่ อีควอไลเซอร์ หมายถึง การทำให้เท่ากัน ในที่นี้จึงหมายถึงการสร้างสัญญาณให้เท่ากับสัญญาณต้นแบบเดิม วัตถุประสงค์ในการใช้อีควอไลเซอร์ แบ่งเป็น 2 แบบ คือ แบบแก้ไขสัญญาณให้ถูกต้อง และ แบบเพิ่มคุณภาพการทำงาน

อีควอไลเซอร์แบบแก้ไขสัญญาณ จะใช้ในระบบบันทึกเสียงในอุปกรณ์เสียงระบบดิจิตอล ตัวอย่างการใช้งานคือ การใช้วงจรพีเอมพีในการบันทึกเทป ซึ่งขบวนการนี้มีมานานแล้วเพื่อชดเชยพลังงานที่ความถี่บางช่วงที่สูญเสียในกระบวนการบันทึก

อีควอไลเซอร์แบบเพิ่มคุณภาพสัญญาณ จะเป็นที่รู้จักกันทั่วไปในงานบันทึกดนตรีใช้เพื่อเน้นหรือเสริมแต่งเสียงดนตรีบางชนิดเป็นพิเศษ ตัวอย่างการใช้งานคือ กรณีที่ต้องการบันทึกเสียงดนตรี 2 ชั้นลงในเทร็คเดียวกันเราอาจใช้อีควอไลเซอร์ก่อนการบันทึกเพื่อเน้นเสียงดนตรีหนึ่งๆ ให้ดูชัดเจนยิ่งขึ้น เช่น เสียงกลอง เป็นต้น

ช่วงความถี่ต่างๆที่อีควอไลเซอร์ใช้จะมีมาตรฐานเป็นตัวกำหนดเราเรียกมาตรฐานนี้ว่า ISO (International Standard Organization) แสดงค่าดังตารางที่ 2.1 อย่างไรก็ตามช่วงความถี่อาจเปลี่ยนได้ตามการใช้งาน เช่น ในเครื่องช่วยฟัง ซึ่งผู้ป่วยที่มีปัญหาด้านการได้ยินอาจต้องการความละเอียดในกระปรับรัดขยายสำหรับบางช่วงความถี่ เช่น 1 kHz – 4 kHz จึงจำเป็นต้องแบ่งช่วงความถี่ย่อยมากเป็นพิเศษ



รูปที่ 2.1 บล็อกไดอะแกรมของอีควอไลเซอร์

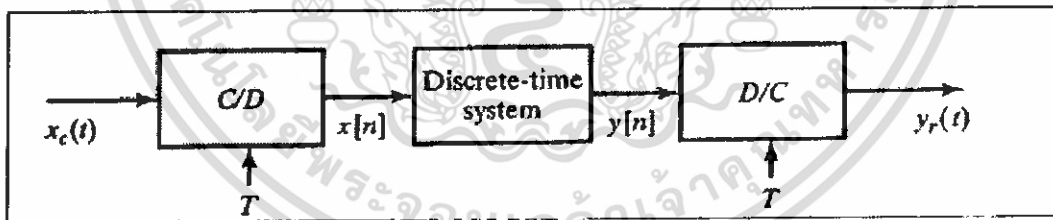
ความถี่กลาง (Hz)	ช่วงความถี่ 1/3 ออกเตฟ (Hz)	ช่วงความถี่ 1 ออกเตฟ (Hz)
20	17.8-22.4	
25	22.4-28.2	
31.5	28.2-35.5	22.4-44.7
40	35.5-44.7	
50	44.7-56.2	
63	56.2-70.8	44.7-89.1
80	70.8-89.1	
100	89.1-112	
125	112-141	89.1-178
160	141-178	
200	178-224	
250	224-282	178-355
315	282-355	
400	355-447	
500	447-562	355-708
630	562-708	
800	708-891	
1000	891-1120	708-1410
1250	1120-1410	
1600	1410-1780	
2000	1780-2240	1410-2820
2500	2240-2820	
3150	2820-3550	
4000	3550-4470	2820-5620
5000	4470-5620	
6300	5620-7080	
8000	7080-8910	5620-11200
10000	8910-11200	
12500	11200-14100	
16000	14100-17800	11200-22400
20000	17800-22400	

ตารางที่ 2.1 แสดงช่วงความถี่ออกเตฟตามมาตรฐานของ ISO

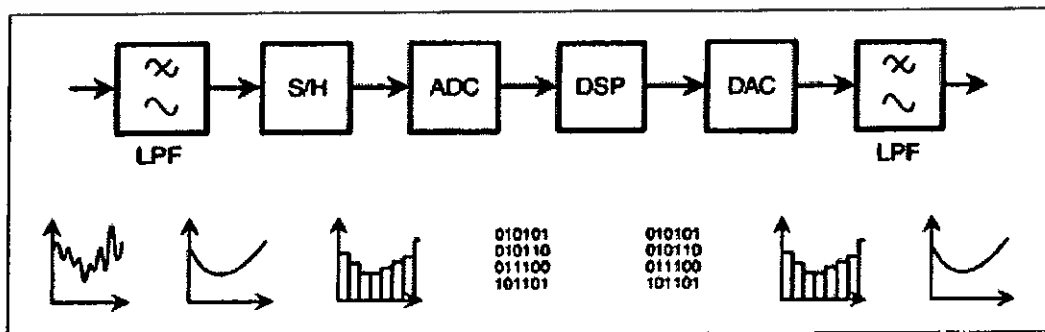
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ตัวกรองเชิงเลข

โดยทั่วไปสัญญาณไฟฟ้าที่เกิดขึ้นจากตัวตรวจจับ (Sensor) ส่วนใหญ่มักจะเป็นสัญญาณอนาล็อก ตัวอย่างเช่น สัญญาณเสียงที่ได้จากไมโครโฟน สัญญาณรูปที่ได้จากกล้องวิดีโอ แรงดันไฟฟ้าที่ได้จากตัวตรวจจับอุณหภูมิ สัญญาณทางการแพทย์ซึ่งสัญญาณที่ได้กล่าวมาแล้วนี้ เป็นสัญญาณอนาล็อกหรือสัญญาณที่มีความต่อเนื่องทางเวลา (Continuous Time Signal) ทั้งสิ้น ดังนั้นถ้าจะทำการวิเคราะห์สัญญาณเหล่านี้ด้วยระบบประมวลผลสัญญาณดิจิทัลจำเป็นต้องแปลงสัญญาณอนาล็อกหรือสัญญาณที่ต่อเนื่องทางเวลาให้เป็นสัญญาณดิจิทัลหรือสัญญาณที่ไม่ต่อเนื่องทางเวลา (Discrete Time Signal) โดยใช้วงจรแปลงอนาล็อกเป็นดิจิทัล (Analog to Digital Converter) จากนั้นข้อมูลจะถูกส่งเข้าไปยังระบบประมวลผลสัญญาณแบบไม่ต่อเนื่องทางเวลา (Discrete Time System) เพื่อทำการคำนวณผลที่ได้จากการคำนวณก็จะถูกแปลงกลับให้เป็นสัญญาณอนาล็อก โดยวงจรแปลงดิจิทัลเป็นอนาล็อกดังแสดงดังรูปที่ 2.2 แต่ในการใช้งานจริงการป้อนสัญญาณที่ต่อเนื่องเข้าไปยังวงจรแปลงอนาล็อกเป็นดิจิทัลอาจเกิดข้อผิดพลาดเนื่องจากการสุ่มสัญญาณ โดยเฉพาะสัญญาณที่มีความถี่สูงความผิดพลาดนี้เรียกว่าการเกิด Aliasing การแก้ไขสามารถทำได้โดยการต่อ Anti-aliasing Filter (Low-pass Filter) เข้าทางด้าน Input และ ทางด้าน Output ก็เช่นเดียวกันสัญญาณที่ได้จากวงจรแปลงดิจิทัลเป็นอนาล็อกจำเป็นต้องนำมาผ่าน Reconstruction Filter (Low-pass Filter) เพื่อปรับสัญญาณให้เรียงดังแสดงในรูปที่ 2.2



รูปที่ 2.2 บล็อกไดอะแกรมของระบบประมวลผลสัญญาณดิจิทัล

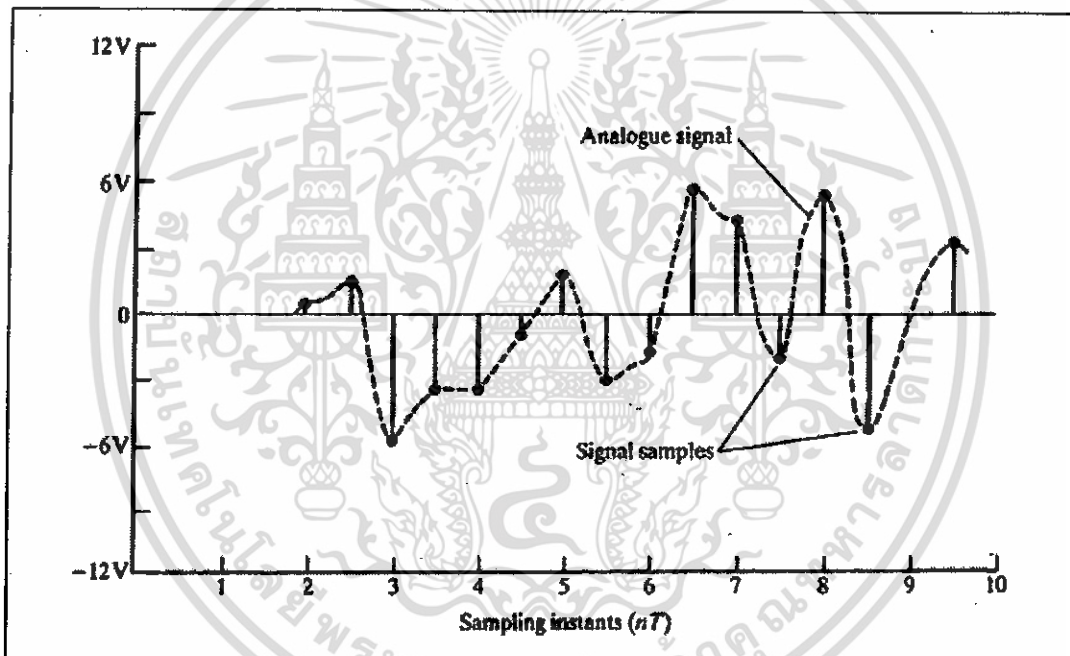


รูปที่ 2.3 บล็อกไดอะแกรมของระบบประมวลผลสัญญาณดิจิทัลที่สมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 สัญญาณแบบไม่ต่อเนื่องทางเวลา

ในระบบประมวลผลสัญญาณแบบไม่ต่อเนื่องสัญญาณที่ใช้ในระบบก็จะเป็นสัญญาณที่ไม่ต่อเนื่องเช่นเดียวกัน ดังที่ได้กล่าวมาแล้วซึ่งสัญญาณที่ไม่ต่อเนื่องนี้จะมีลักษณะดังรูปที่ 2.4 โดยจะเห็นว่าสัญญาณที่ไม่ต่อเนื่องก็คือสัญญาณค่าหนึ่ง ณ เวลาหนึ่งบนสัญญาณที่ต่อเนื่องแต่จะมีระยะเวลาห่างที่เท่ากันในแต่ละจุด เรียกระยะเวลาห่างที่เท่ากันนี้ว่า เวลาการสุ่ม(Sampling Time), อัตราการสุ่ม (Sampling Rate)หรือช่วงการสุ่ม (Sampling Periods) สัญญาณที่ไม่ต่อเนื่องสามารถเรียกเป็นลำดับสัญญาณ (Sequence Signal) ก็ได้ และสามารถเขียนแทนได้ด้วยสมการคณิตศาสตร์ได้หลายๆแบบดังต่อไปนี้



รูปที่ 2.4 สัญญาณต่อเนื่องและสัญญาณไม่ต่อเนื่อง

เขียนในรูปของฟังก์ชัน (Functional Representation)

$$x(n) = \begin{cases} 1, & n = 1, 3 \\ 4, & n = 2 \\ 0, & n > 4 \end{cases} \quad (2.1)$$

เขียนในรูปการแจกแจง (Tabular Representation)

n	...	-2	-1	0	1	2	3	4	...
$x(n)$...	0	0	0	1	4	1	0	...

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เขียนในรูปลำดับ (Sequence Representation)

$$x(n) = \{\dots, 0, 0, 1, 4, 1, 0, 0, \dots\} \quad (2.2)$$

$$x(n) = \{0, 1, 4, 1, 0, 0, \dots\} \quad (2.3)$$

$$x(n) = \{3, -1, -2, 5, 0, 4, -1\} \quad (2.4)$$

$$x(n) = \{0, 1, 4, 1\} \quad (2.5)$$

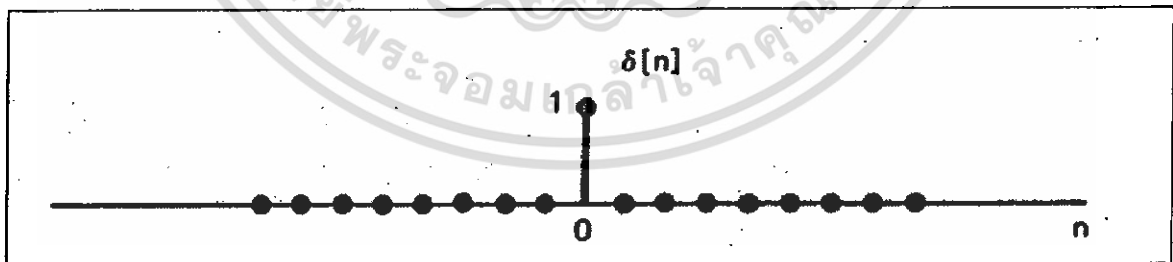
สมการที่ 2.2 และ 2.3 เป็นลำดับแบบไม่มีจุดจบ (Infinite - duration Signal or Sequence)

ส่วนสมการที่ 2.4 และ 2.5 เป็นลำดับแบบมีจุดจบ (Finite - duration Signal or Sequence)

2.2.1.1 ฟังก์ชันอิมพัลส์หนึ่งหน่วย

นิยามดังสมการที่ 2.6 และมีรูปสัญลักษณ์ดังรูปที่ 2.4

$$\delta(n) = \begin{cases} 0, & n \neq 0 \\ 1, & n = 0 \end{cases} \quad (2.6)$$

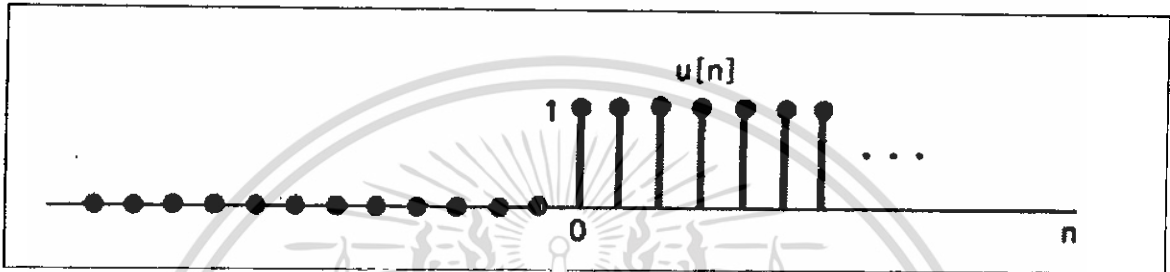


รูปที่ 2.5 ฟังก์ชันอิมพัลส์หนึ่งหน่วย

2.2.1.2 ฟังก์ชันขั้นหนึ่งหน่วย

มีนิยามดังสมการที่ 2.7 และมีรูปสัญญาณดังรูปที่ 2.6

$$u(n) = \begin{cases} 0, & n < 0 \\ 1, & n \geq 0 \end{cases} \quad (2.7)$$

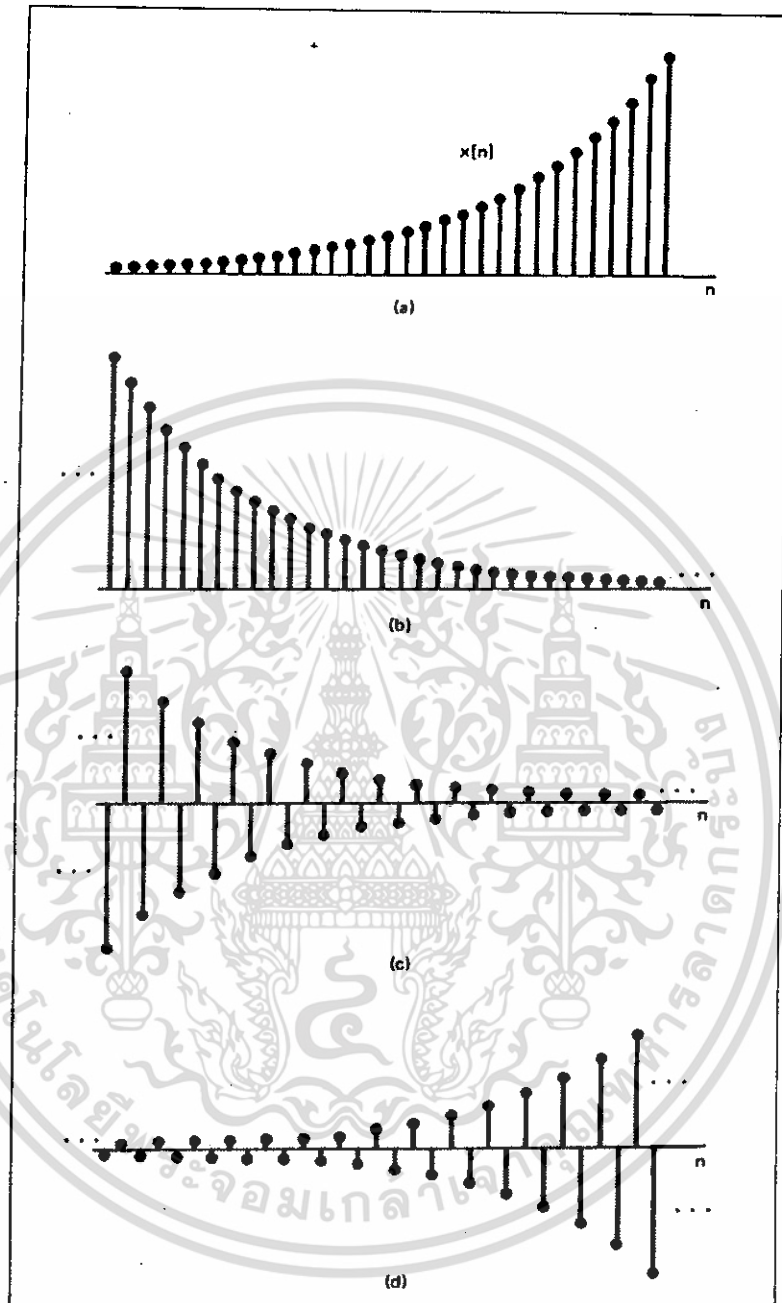


รูปที่ 2.6 ฟังก์ชันขั้นหนึ่งหน่วย

2.2.1.3 ฟังก์ชันเอกซ์โพเนนเชียล

มีนิยามดังสมการที่ 2.8 และมีรูปสัญญาณดังรูปที่ 2.7

$$x(n) = \alpha^n \quad (2.8)$$



รูปที่ 2.7 พังก์ชันเอกซ์โพเนนเชียล $x(n) = A\alpha^n$

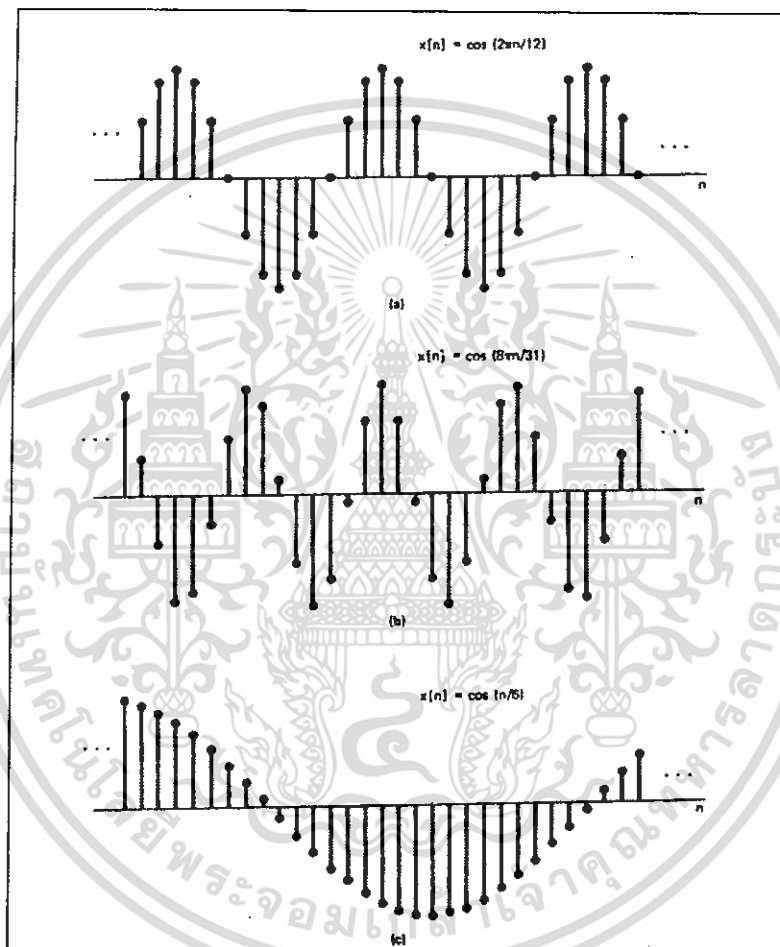
- (a) $\alpha > 1$
- (b) $0 < \alpha < 1$
- (c) $-1 < \alpha < 0$
- (d) $\alpha < -1$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1.4 ฟังก์ชันซายน์

มีนิยามดังสมการที่ 2.9 และมีรูปสัญญาณดังรูปที่ 2.8

$$x(n) = A \cos(\omega_0 n + \phi) \quad (2.9)$$



รูปที่ 2.8 ฟังก์ชันซายน์

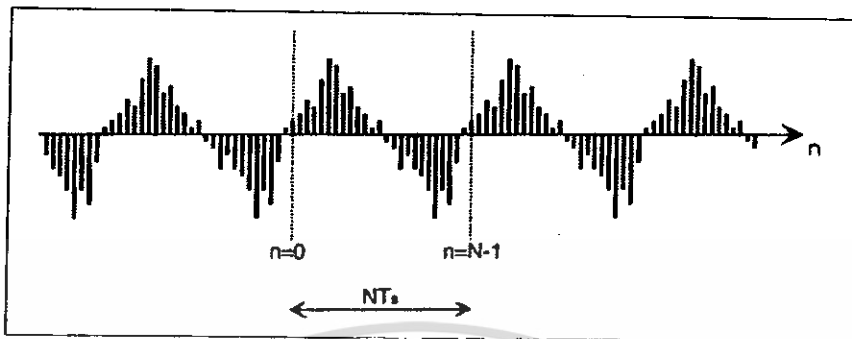
2.2.1.5 สัญญาณแบบมีคาบและสัญญาณแบบไม่มีคาบ

สัญญาณแบบมีคาบมีนิยามดังสมการที่ 2.10

$$x(n + N) = x(n) \quad (2.10)$$

เมื่อ N คือ Fundamental Period และลำดับไม่เป็นดังสมการที่ 2.10 จะเป็นลำดับแบบไม่มีคาบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



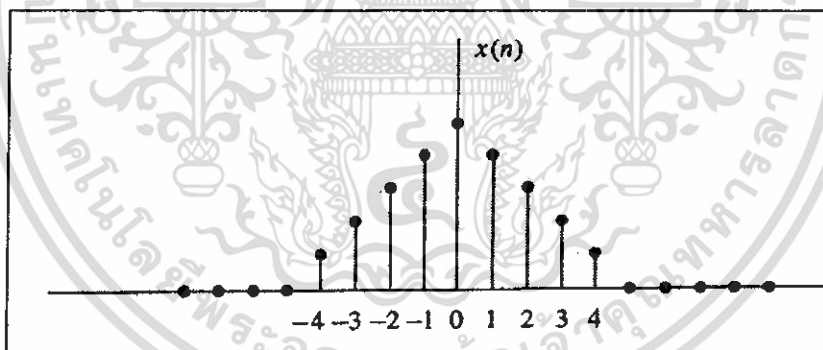
รูปที่ 2.9 สัญญาณแบบมีคาบ

2.2.1.6 สัญญาณสมมาตร (even) และไม่สมมาตร (odd)

ลำดับสัญญาณที่สมมาตรกัน (Symmetric) มีนิยามดังสมการที่ 2.11 มีรูปสัญญาณดังรูปที่

2.9

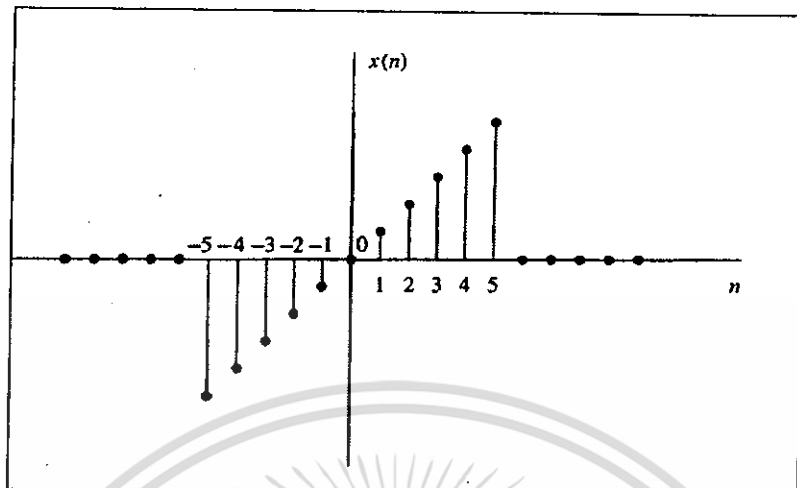
$$x(-n) = x(n) \tag{2.11}$$



รูปที่ 2.10 สัญญาณสมมาตร

ลำดับสัญญาณที่ไม่สมมาตรกัน (Antisymmetric) มีนิยามดังสมการที่ 2.12 มีรูปสัญญาณดังรูปที่ 2.11

$$x(-n) = -x(n) \tag{2.12}$$



รูปที่ 2.11 สัญญาณไม่สมมาตร

2.2.1.7 การกระทำของสัญญาณแบบไม่ต่อเนื่องทางเวลา

- การเลื่อนสัญญาณ

อาจเรียกว่าการหน่วงสัญญาณก็ได้เขียนเป็นสมการได้ดังสมการที่ 2.13

$$y(n) = x(n - k) \quad (2.13)$$

เมื่อ k เป็นจำนวนเต็ม

- การบวก การคูณ และการสเกลสัญญาณแบบไม่ต่อเนื่อง
- การบวกสัญญาณ 2 สัญญาณ หรือ k สัญญาณเข้าด้วยกัน ได้ดังสมการที่ 2.14 และสมการที่ 2.15

$$y(n) = x_1(n) + x_2(n) \quad -\infty < n < \infty \quad (2.14)$$

$$y(n) = x_1(n) + x_2(n) + \dots + x_k(n) \quad -\infty < n < \infty \quad (2.15)$$

- การคูณสัญญาณ 2 สัญญาณ หรือ k สัญญาณเข้าด้วยกันดังสมการที่ 2.16 และสมการที่ 2.17

$$y(n) = x_1(n)x_2(n) \quad -\infty < n < \infty \quad (2.16)$$

$$y(n) = x_1(n)x_2(n)\dots x_k(n) \quad -\infty < n < \infty \quad (2.17)$$

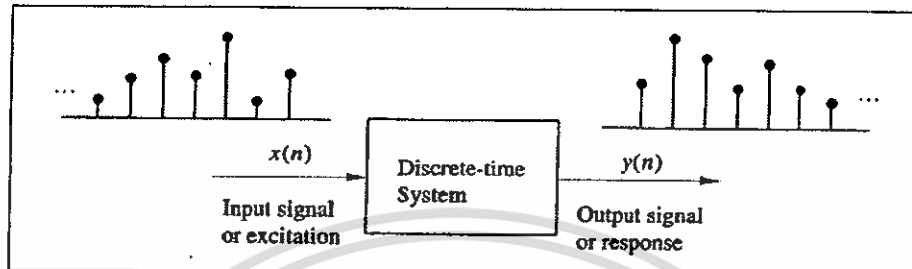
- การสเกลสัญญาณเป็นการปรับขนาดสัญญาณกระทำได้การคูณสัญญาณเข้ากับค่าคงที่ดังสมการที่ 2.18

$$y(n) = Ax(n) \quad -\infty < n < \infty \quad (2.18)$$

เมื่อ A เป็นค่าคงที่

2.2.2 ระบบแบบไม่ต่อเนื่องทางเวลา

ระบบแบบไม่ต่อเนื่องมีบล็อกไดอะแกรมดังรูปที่ 2.12



รูปที่ 2.12 บล็อกไดอะแกรมของระบบแบบไม่ต่อเนื่อง

เมื่อ $x(n)$ เป็น สัญญาณอินพุต หรือ Excitation ของระบบแบบไม่ต่อเนื่อง
 $y(n)$ เป็น สัญญาณเอาต์พุต หรือ Response ของระบบแบบไม่ต่อเนื่อง
 จากรูปที่ 2.12 สามารถเขียนความสัมพันธ์ระหว่างอินพุตกับเอาต์พุตได้ดังสมการที่ 2.19

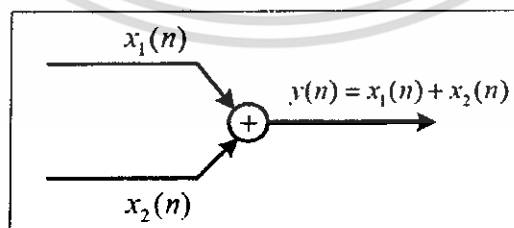
$$y(n) \equiv \tau[x(n)] \quad (2.19)$$

เมื่อ τ เป็นตัวกระทำ (Operator)

2.2.2.1 บล็อกไดอะแกรมที่ใช้แสดงในระบบไม่ต่อเนื่อง

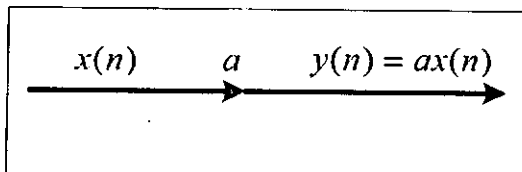
ในระบบแบบไม่ต่อเนื่องสามารถแทนได้ด้วยไดอะแกรมดังต่อไปนี้

- ตัวบวก (Adder) ใช้สำหรับบวกสัญญาณเข้าด้วยกันดังรูปที่ 2.13



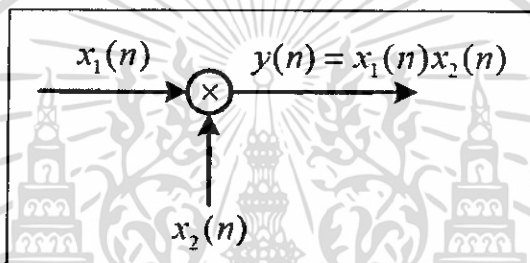
รูปที่ 2.13 บล็อกไดอะแกรมการบวกในระบบแบบไม่ต่อเนื่อง

- ตัวคูณคงที่ (Constant Multiplier) ใช้คูณค่าคงที่กับสัญญาณดังรูปที่ 2.14



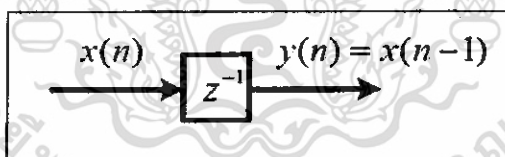
รูปที่ 2.14 บล็อกโคอะแกรมตัวคูณคงที่ในระบบแบบไม่ต่อเนื่อง

- ตัวคูณสัญญาณ (Signal Multiplier) ใช้สำหรับคูณสัญญาณเข้าด้วยกันดังรูปที่ 2.15



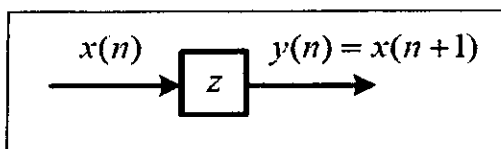
รูปที่ 2.15 บล็อกโคอะแกรมตัวคูณสัญญาณในระบบแบบไม่ต่อเนื่อง

- ตัวหน่วง (Delay Element) ใช้สำหรับหน่วงสัญญาณดังรูปที่ 2.16



รูปที่ 2.16 บล็อกโคอะแกรมตัวหน่วงในระบบไม่ต่อเนื่อง

- ตัวล้าหน้า (Advance Element) ใช้สำหรับกำหนดค่าล่วงหน้าให้กับระบบดังรูปที่ 2.17



รูปที่ 2.17 บล็อกโคอะแกรมตัวล้าหน้าในระบบไม่ต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2.2 ชนิดของระบบแบบไม่ต่อเนื่อง

- ระบบแบบ Static เป็นระบบที่ไม่มีหน่วยความจำมีสมการของระบบดังสมการที่ 2.20 และสมการที่ 2.21

$$y(n) = ax(n) \quad (2.20)$$

$$y(n) = nx(n) + bx^3(n) \quad (2.21)$$

คือ ไม่ใช่ค่าอินพุตหรือเอาต์พุต ของหน่วยเวลาก่อนหน้า

- ระบบแบบ Dynamic เป็นระบบที่มีหน่วยความจำนั้นหมายถึงเป็นระบบที่สามารถบันทึกค่าอินพุตหรือเอาต์พุตที่เวลาผ่านไปแล้วได้มีสมการของระบบดังสมการที่ 2.22 - 2.24

$$y(n) = x(n) + 3x(n-1) \quad (2.22)$$

$$y(n) = \sum_{k=0}^n x(n-k) \quad (2.23)$$

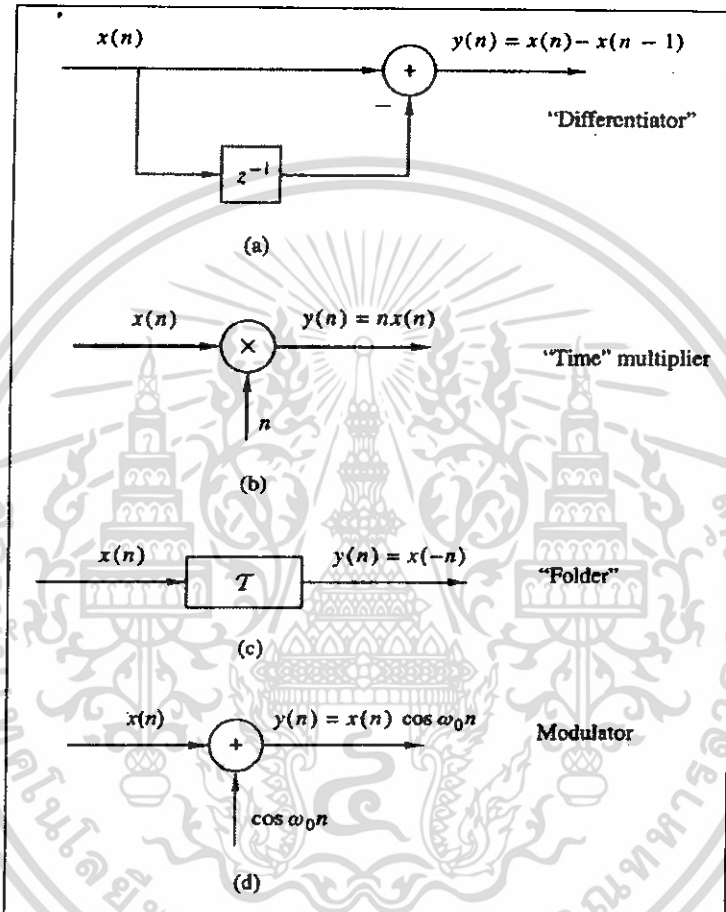
$$y(n) = \sum_{k=0}^{\infty} x(n-k) \quad (2.24)$$

- ระบบแบบไม่แปรตามเวลา (Time-invariant System) กำหนดให้ τ เป็นระบบไม่ต่อเนื่อง $x(n)$ เป็นอินพุตของระบบ $y(n)$ เป็นเอาต์พุตของระบบดังสมการที่ 2.25 ถ้ามีการหน่วงอินพุตเป็น $x(n-k)$ ถ้าเอาต์พุตที่ได้ถูกหน่วงไปเป็น $y(n-k)$ เช่นเดียวกับอินพุตดังสมการที่ 2.26 สามารถสรุปได้ว่าระบบไม่ต่อเนื่อง τ เป็นระบบไม่ต่อเนื่องที่แปรตามเวลา

$$x(n) \xrightarrow{\tau} y(n) \quad (2.25)$$

$$x(n-k) \xrightarrow{\tau} y(n-k) \quad (2.26)$$

และระบบที่อินพุตถูกหน่วงไป $x(n-k)$ แต่ได้เอาที่พุทของระบบไม่เท่ากับ $y(n-k)$ ระบบนั้นเป็นระบบที่แปรตามเวลา (Time-variant System หรือ Time-varying System) รูปที่ 2.18 แสดงตัวอย่างระบบไม่ต่อเนื่องที่แปรตามเวลาและไม่แปรตามเวลา



รูปที่ 2.18 ระบบไม่ต่อเนื่องที่แปรตามเวลาและไม่แปรตามเวลา

(a) ระบบแปรตามเวลา

(b) - (d) ระบบที่ไม่แปรตามเวลา

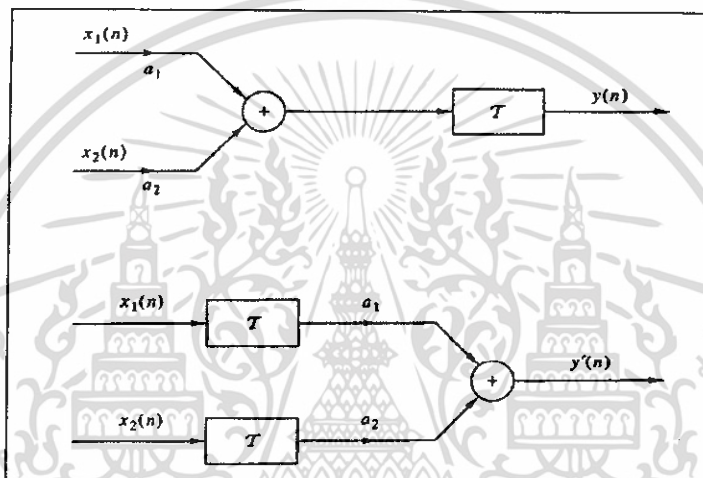
72090

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ระบบแบบเชิงเส้นและไม่เชิงเส้น

ระบบไม่ต่อเนื่องแบบเชิงเส้นจะต้องมีคุณสมบัติสอดคล้องกับหลักการของ การทับซ้อน (Superposition) ดังสมการที่ 2.27 หรืออธิบายได้ดังรูปที่ 2.19 และระบบที่ไม่มีคุณสมบัติสอดคล้องกับหลักการของ Superposition จะเป็นระบบแบบไม่เชิงเส้น

$$\tau[a_1x_1(n) + a_2x_2(n)] = a_1\tau[x_1(n)] + a_2\tau[x_2(n)] \quad (2.27)$$



รูปที่ 2.19 แสดงระบบ τ เป็นระบบที่เชิงเส้น

และระบบที่ไม่มีคุณสมบัติสอดคล้องกับหลักการของการทับซ้อนจะเป็นระบบแบบไม่เชิงเส้น

- ระบบแบบ Causal และแบบ Non-causal

ระบบแบบ Causal เป็นระบบที่สร้างได้จริงมีสมการของระบบดังสมการที่ 2.29

$$y(n) = F[x(n), x(n-1), x(n-2), \dots, x(n-k), \dots, y(n-1), y(n-2), \dots, y(n-k)] \quad (2.28)$$

$$y(n) = 2x(n) + 4.5x(n-2) - 3y(n-1) + 0.5y(n-2) \quad (2.29)$$

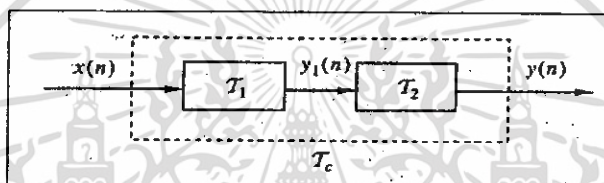
จะเห็นได้ว่าระบบแบบ Causal จะประกอบด้วยเทอมอินพุตปัจจุบันอินพุตในอดีต หรือ เอาท์พุตในอดีตเท่านั้น ระบบที่ไม่เป็นดังสมการที่ 2.29 จะเป็นระบบแบบ Non-causal

- ระบบแบบเสถียร (Stable) และระบบแบบ ไม่เสถียร (Unstable)

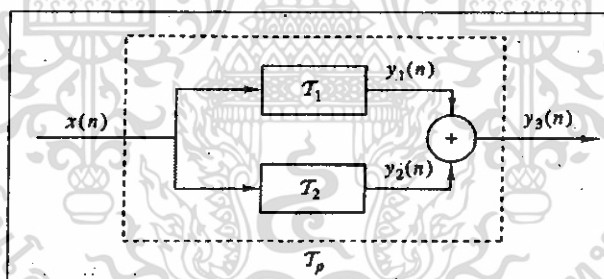
ระบบที่มีเอาต์พุตขึ้นอยู่กับอินพุตจะเป็นระบบแบบเสถียรส่วนระบบที่มีเอาต์พุต ไม่ขึ้นอยู่กับอินพุตจะเป็นระบบแบบไม่เสถียร

2.2.2.3 การเชื่อมต่อระบบไม่ต่อเนื่อง (Interconnection of Discrete-time Systems)

กระทำได้ 2 ลักษณะคือ แบบอนุกรม (Cascade) ดังแสดงดังรูปที่ 2.20 และแบบขนาน (Parallel) ดังแสดงดังรูปที่ 2.21



รูปที่ 2.20 การเชื่อมต่อระบบไม่ต่อเนื่องแบบอนุกรม



รูปที่ 2.21 การเชื่อมต่อระบบไม่ต่อเนื่องแบบขนาน

ในการเชื่อมต่อระบบดังรูปที่ 2.20 เขียนเป็นสมการได้ดังสมการที่ 2.30 – 2.32

$$y_1(n) = \tau_1[x(n)] \quad (2.30)$$

$$y(n) = \tau_2[y_1(n)] \quad (2.31)$$

แทนสมการที่ 2.30 ลงในสมการที่ 2.31 จะได้สมการ $y(n)$ ดังสมการที่ 2.32

$$y(n) = \tau_2\{\tau_1[x(n)]\} \quad (2.32)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดให้ $\tau_c = \tau_1 \tau_2$ จะได้ $y(n)$ ดังสมการที่ 2.33

$$y(n) = \tau_c [x(n)] \quad (2.33)$$

และถ้ามีการสลับที่ระบบ τ_1 กับ τ_2 $y(n)$ จะมีค่าเปลี่ยนแปลงไปดังนั้นจึงสรุปได้ว่า

$$\tau_1 \tau_2 \neq \tau_2 \tau_1 \quad (2.34)$$

ส่วนการเชื่อมต่อแบบขนานเขียนเป็นสมการได้ดังสมการที่ 2.36

$$\begin{aligned} y_3(n) &= y_1(n) + y_2(n) \\ &= \tau_1 [x(n)] + \tau_2 [x(n)] \\ &= (\tau_1 + \tau_2) [x(n)] \\ &= \tau_p [x(n)] \end{aligned} \quad (2.35)$$

เมื่อ $\tau_p = \tau_1 + \tau_2$

2.2.2.4 การวิเคราะห์ระบบไม่ต่อเนื่องแบบเชิงเส้นและแปรตามเวลา

- Resolution of a Discrete-time Signal into Impulses

สัญญาณแบบไม่ต่อเนื่องใดๆสามารถเขียนแทนในรูปของ Impulse ได้เช่นดังรูปที่ 2.22 (b) เขียนเป็นสมการได้ดังสมการที่ 2.36

$$x(n] = \delta(n - k) \quad (2.36)$$

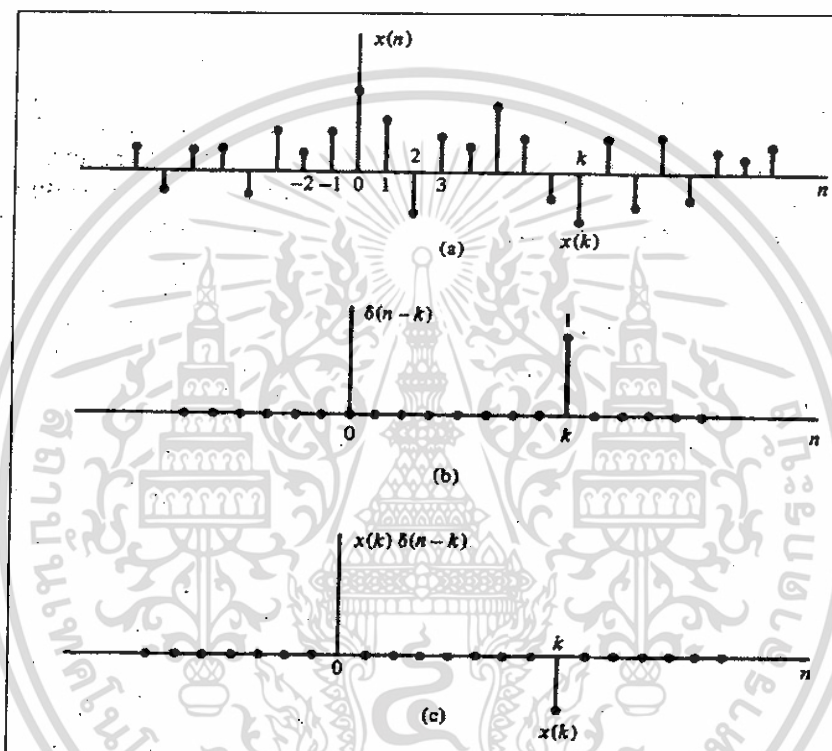
และรูปที่ 2.22 (c) เขียนได้ดังสมการที่ 2.37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$x(n) = x(k)\delta(n - k) \quad (2.37)$$

ดังนั้นรูปที่ 2.22 (a) เขียนแทนได้ดังสมการที่ 2.38

$$x(n) = \sum_{k=-\infty}^{\infty} x(k)\delta(n - k) \quad (2.38)$$



รูปที่ 2.22 การแทนสัญญาณแบบไม่ต่อเนื่องในรูป Impulse

- Convolution Sum

เป็นการผสมผสานระหว่างสัญญาณแบบไม่ต่อเนื่อง $x(n)$ กับผลตอบสนองทางอิมพัลส์ของระบบไม่ต่อเนื่อง $h(n)$ เขียนได้ดังสมการที่ 2.39

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n - k) \quad (2.39)$$

เมื่อ $h(k)$ คือ ผลตอบสนองทางอิมพัลส์ของระบบไม่ต่อเนื่อง

$x(n)$ คือ สัญญาณแบบไม่ต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นได้สมการสำหรับระบบแบบ Causal การทำ Convolution จะทำตามสมการที่ 2.40

$$y(n) = \sum_{k=0}^n h(k)x(n-k) \quad (2.40)$$

นอกจากนี้สมการ Convolution สามารถเขียนได้อีกแบบหนึ่งดังสมการที่ 2.41

$$y(n) = h(n) * x(n) \quad (2.41)$$

- Commutative law

$$x(n) * h(n) = h(n) * x(n) \quad (2.42)$$

เขียนเป็นบล็อกไดอะแกรมได้ดังรูปที่ 2.23



รูปที่ 2.23 คุณสมบัติการสลับที่

$$y(n) = x(n) * h(n) \quad (2.43)$$

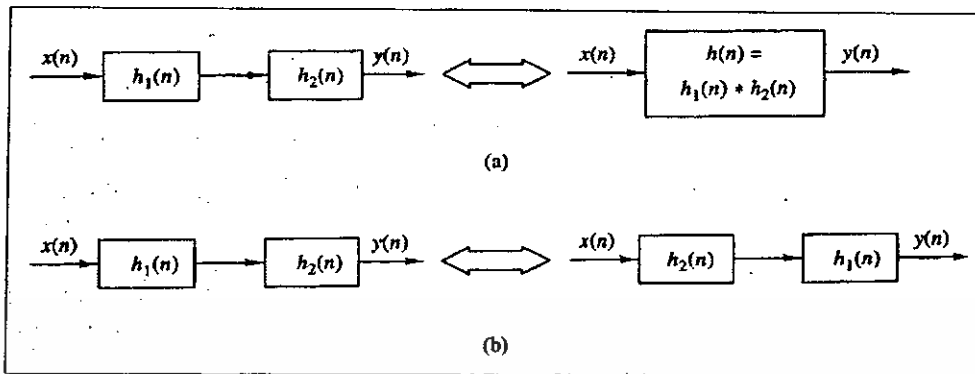
$$y(n) = h(n) * x(n) \quad (2.44)$$

- Associative law

$$[x(n) * h_1(n)] * h_2(n) = x(n) * [h_1(n) * h_2(n)] \quad (2.45)$$

เขียนเป็นบล็อกไดอะแกรมได้ดังรูปที่ 2.24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

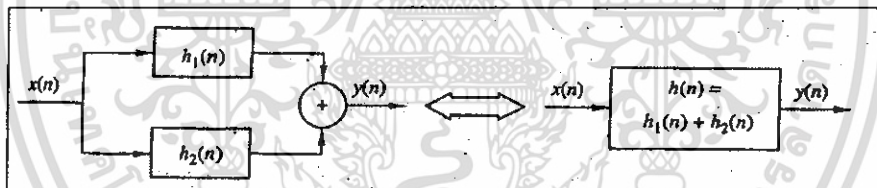


รูปที่ 2.24 คุณสมบัติการจับกลุ่ม

- Distributive Law

$$x(n) * [h_1(n) + h_2(n)] = x(n) * h_1(n) + x(n) * h_2(n) \quad (2.46)$$

เขียนเป็นบล็อกไดอะแกรมได้ดังรูปที่ 2.25



รูปที่ 2.25 คุณสมบัติการกระจาย

2.2.3 z-transform

การแปลงแซด (z-transform) ของสัญญาณที่ไม่ต่อเนื่องสามารถนิยามอยู่ในรูปของอนุกรมอนันต์ (Power Series) ได้ดังสมการที่ 2.47 เรียกสมการที่ 2.47 นี้ว่า การแปลงแซดแบบตรง (Direct z-transform)

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n} \quad (2.47)$$

เมื่อ z เป็นตัวแปรเชิงซ้อน (Complex Variable)

$x(n)$ เป็นลำดับสัญญาณ (Sequence Signal)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่สำหรับ Causal System การแปลงแซดของสัญญาณที่ไม่ต่อเนื่องสามารถเขียนได้ดังสมการที่ 2.48 เรียกสมการที่ 2.48 นี้ว่า One-side z-transform

$$X(z) = \sum_{n=0}^{\infty} x(n)z^{-n} \quad (2.48)$$

การแปลงแซดของสัญญาณแบบไม่ต่อเนื่องสามารถเขียนแทนได้อีกรูปแบบหนึ่งดังสมการที่ 2.49

$$X(z) = z\{x(n)\} \quad (2.49)$$

และจากสมการที่ 2.52 และ 2.53 จะเห็นได้ว่าผลการแปลงแซด ก็คืออนุกรมอนันต์ ดังนั้นจะมีค่า z ที่ทำให้ $X(z)$ หาคำตอบได้เรียกค่า z นี้ว่า Region of Convergence (ROC) โดยค่า z นี้จะมีค่าเป็นจำนวนเชิงซ้อน

Entry Number	Discrete-time Sequence	z - Transform $X(z)$	Region of Convergence of $X(z)$
1	$\delta[n]$	1	All z
2	$u[n]$	$\frac{1}{1-z^{-1}}$	$ z > 1$
3	$-u[-n-1]$	$\frac{1}{1-z^{-1}}$	$ z < 1$
4	$\delta[n-m]$	z^{-m}	All z except 0 (if $m > 0$) or ∞ (if $m < 0$)
5	$a^n u[n]$	$\frac{1}{1-az^{-1}}$	$ z > a $
6	$-a^n u[-n-1]$	$\frac{1}{1-az^{-1}}$	$ z < a $
7	$na^n u[n]$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$ z > a $
8	$-na^n u[-n-1]$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$ z < a $
9	$[\cos \omega_0 n] u[n]$	$\frac{1 - [\cos \omega_0] z^{-1}}{1 - [2 \cos \omega_0] z^{-1} + z^{-2}}$	$ z > 1$
10	$[\sin \omega_0 n] u[n]$	$\frac{[\sin \omega_0] z^{-1}}{1 - [2 \cos \omega_0] z^{-1} + z^{-2}}$	$ z > 1$
11	$[r^n \cos \omega_0 n] u[n]$	$\frac{1 - [r \cos \omega_0] z^{-1}}{1 - [2r \cos \omega_0] z^{-1} + r^2 z^{-2}}$	$ z > r$
12	$[r^n \sin \omega_0 n] u[n]$	$\frac{[r \sin \omega_0] z^{-1}}{1 - [2r \cos \omega_0] z^{-1} + r^2 z^{-2}}$	$ z > r$
13	$\begin{cases} a^n, & 0 \leq n \leq N-1 \\ 0, & \text{otherwise} \end{cases}$	$\frac{1 - a^N z^{-N}}{1 - az^{-1}}$	$ z > 0$

ตารางที่ 2.2 z -transform ของสัญญาณแบบไม่ต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 ตัวกรองดิจิทัล

ตัวกรองดิจิทัลแบ่งออกได้เป็น 2 ชนิด คือ

- ตัวกรองดิจิทัลแบบเอฟไออาร์ (FIR: Finite Impulse Response Filter)
- ตัวกรองดิจิทัลแบบไอไออาร์ (IIR: Infinite Impulse Response Filter)

ซึ่งสามารถเปรียบเทียบกัน ได้ดังตารางที่ 2.3

FIR	IIR
เสถียร	ไม่เสถียร
Transition band กว้าง	Transition band แคบ
Delay มาก	Delay น้อย
ผลตอบสนองเชิงเฟสเป็นเชิงเส้น	ผลตอบสนองเชิงเฟสผิดเพี้ยนสูง
สัญญาณรบกวนจากการบิดเบือนน้อย	สัญญาณรบกวนจากการบิดเบือนมาก
Ripple มาก	Ripple น้อย

ตารางที่ 2.3 เปรียบเทียบตัวกรองเชิงเลข

ในโครงการนี้กล่าวถึงใช้ตัวกรองดิจิทัลแบบ FIR ซึ่งตัวกรองดิจิทัลแบบ FIR มีฟังก์ชันการถ่ายโอน (Transfer Function) มีสมการคุณสมบัติดังสมการที่ 2.50

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) \quad (2.50)$$

เขียนให้อยู่ในแซดโดเมน (z-domain) ได้ดังสมการที่ 2.51

$$y(z) = \sum_{k=0}^{N-1} h(k)x(z)z^{-k} \quad (2.51)$$

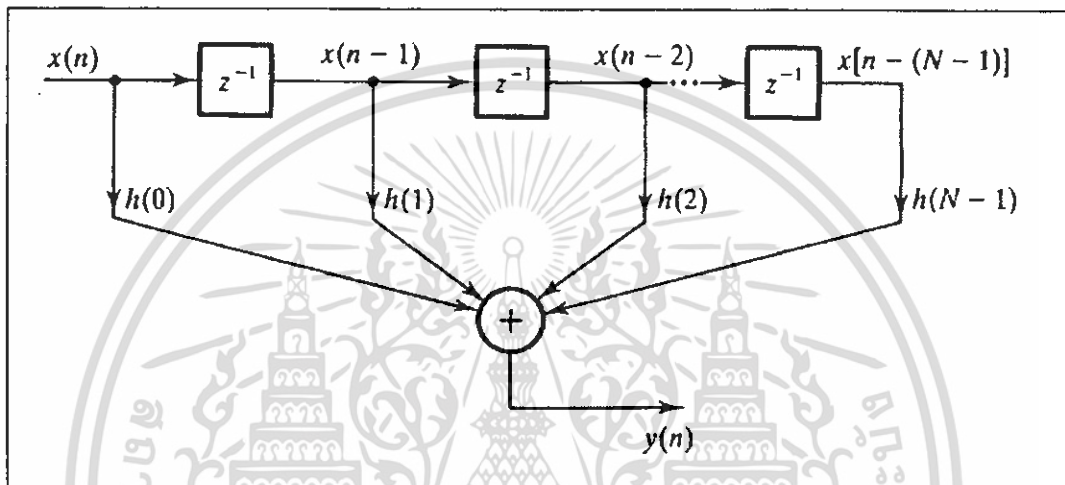
หรือเขียนให้อยู่ในรูปของฟังก์ชันถ่ายโอนจะได้

$$H(z) = \sum_{k=0}^{N-1} h(k)z^{-k} \quad (2.52)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$Y(z) = h(0)x(z) + h(1)x(z)z^{-1} + h(2)x(z)z^{-2} + \dots + h(k)x(z)z^{-k} \quad (2.53)$$

จากสมการที่ 2.94 สามารถนำไปเขียนเป็นโครงสร้างได้ดังรูปที่ 2.26



รูปที่ 2.26 โครงสร้างของตัวกรองเชิงเลข FIR

ตัวกรองเชิงเลข FIR จัดเป็นตัวกรองนอนรีเคอร์ซีฟ (Non Recursive) เนื่องจากไม่มีการป้อนกลับจากทางด้าน Output และนอกจากนี้ตัวกรองเชิงเลข FIR จะมีคุณสมบัติอื่นอีกคือ

1. สามารถสร้างได้ง่ายเมื่อเทียบกับตัวกรองแบบ IIR
2. มีคุณสมบัติการตอบสนองทางเฟสเป็นแบบเชิงเส้น (Linear phase)
3. สัมประสิทธิ์ที่เกิดจากการคำนวณจะมีค่าไม่เกินหนึ่งทำให้สามารถสร้างและทำงานได้ดีบน ตัวประมวลผลแบบจุดทศนิยมคงที่ (Fix-point)
4. ตัวกรองเชิงเลข FIR จะเสถียรภาพ (Stable) แน่นนอน เนื่องจากว่ามีโพลอยู่ที่จุดกำเนิด (Origin) บนระนาบซัด

พิจารณาสมการ

$$y(n) = x(n - k) \quad (2.54)$$

เมื่อทำการแปลงฟูเรียร์ในสมการที่ 2.54 จะได้

$$Y(j\omega) = e^{-j\omega k} X(j\omega) \quad (2.55)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ย้ายข้างจะได้

$$\frac{Y(j\omega)}{X(j\omega)} = H(j\omega) = e^{-j\omega kT} \quad (2.56)$$

จากสมการจะมีค่าขนาด (Magnitude) เท่ากับ 1 และมีเฟสดังสมการที่ 2.57

$$\theta(\omega) = -\omega kT \quad (2.57)$$

และจากสมการที่ 2.57 ซึ่งเป็นสมการของเฟสสามารถนำไปหาค่าหน่วงกลุ่ม (Group Delay) ได้โดยทำการหาอนุพันธ์สมการที่ 2.57 เทียบกับ จะได้สมการของค่าหน่วงกลุ่มดังสมการที่ 2.58

$$d\theta = \frac{d(-\omega kT)}{d\omega} \quad (2.58)$$

$$\Theta = -kT \quad (2.59)$$

จากสมการที่ 2.59 จะเห็นได้ว่าตัวกรองเชิงเลข FIR มีผลตอบสนองทางเฟสเป็นแบบเชิงเส้น ดังนั้นตัวกรองเชิงเลข FIR จึงถูกนำไปใช้งานอย่างกว้างขวาง ดังตัวอย่างเช่น การประมวลผลสัญญาณเสียง การประมวลผลสัญญาณภาพ และการประมวลผลสัญญาณทางการแพทย์ เป็นต้น และถ้ากำหนดสมการผลตอบสนองทางเฟสใหม่โดยกำหนดให้

$$kT = \alpha \quad (2.60)$$

จะได้

$$\theta(\omega) = -\alpha\omega \quad (2.61)$$

หรือถ้าค่าของผลตอบสนองทางเฟสเท่ากับ

$$\theta(\omega) = \beta - \alpha\omega \quad (2.62)$$

เมื่อ β เป็นค่าคงที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าตัวกรองมีทั้งผลการตอบสนองทางเฟสและค่าหน่วยกลุ่มเป็นแบบเชิงเส้นตามสมการที่ 2.61 จะให้ค่าผลตอบสนองอิมพัลส์ของตัวกรองเป็นแบบสมมาตรบวก (Positive Symmetry) ดังสมการที่ 2.63 และ 2.64 ซึ่งผลการตอบสนองทางเฟสจะเป็นฟังก์ชันของความยาวตัวกรอง (Filter Length)

$$h(n) = h(N - n - 1) \text{ ที่ } n = 0, 1, \dots, (N - 1) / 2 \text{ และ } n \text{ เป็นคู่} \quad (2.63)$$

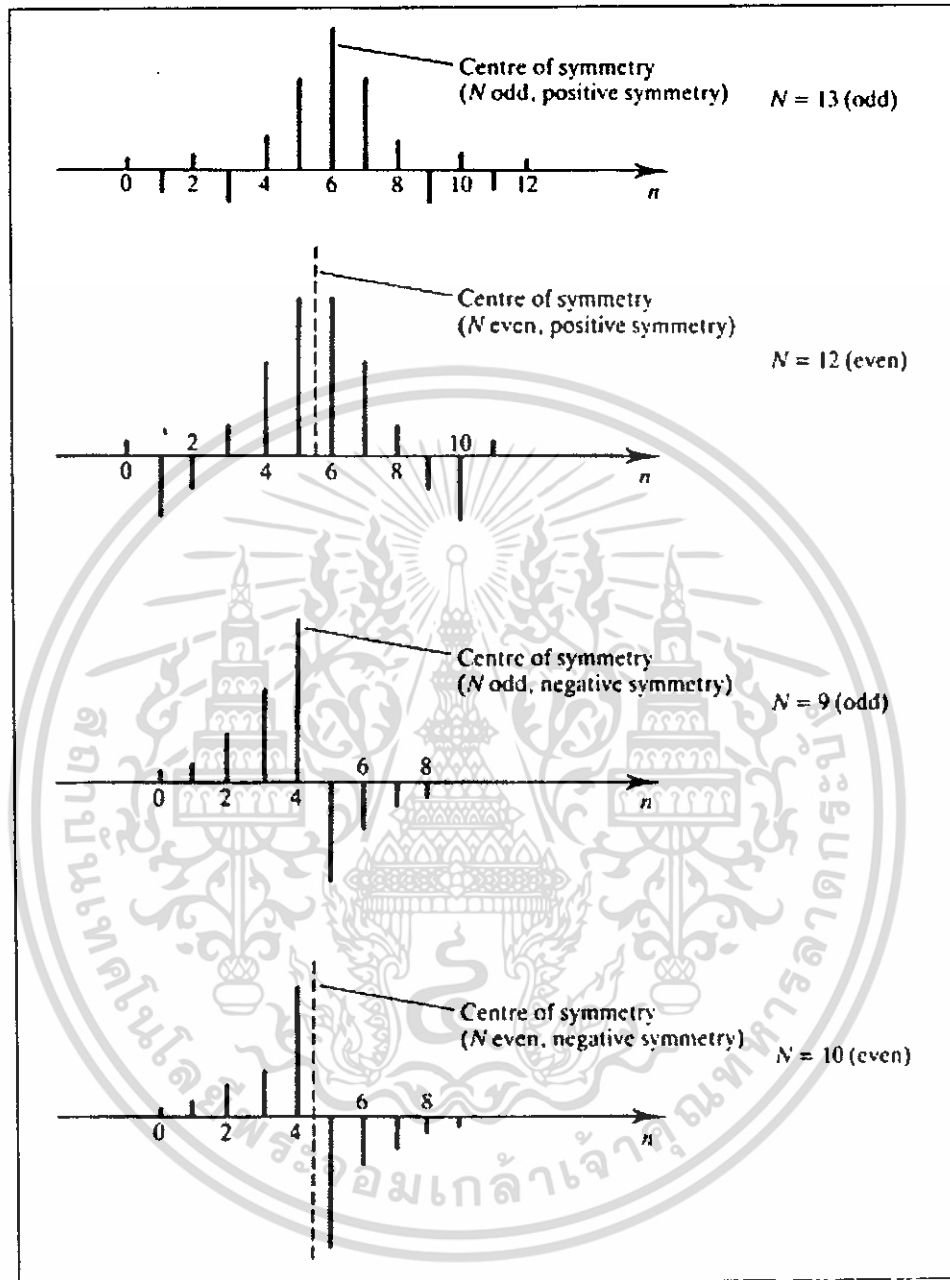
$$\alpha = \frac{(N - 1)}{2} \text{ ที่ } n = 0, 1, \dots, (\frac{N}{2} - 1) \text{ และ } n \text{ เป็นคู่} \quad (2.64)$$

และถ้าฟิลเตอร์มีผลการตอบสนองทางเฟสดังสมการที่ 2.62 จะได้ผลตอบสนองอิมพัลส์ของตัวกรองเป็นแบบสมมาตรลบ (Negative Symmetry) ดังสมการที่ 2.65 และ 2.66

$$h(n) = -h(N - n - 1) \quad (2.65)$$

$$\alpha = \frac{(N - 1)}{2} \quad (2.66)$$

โดย N ในสมการที่ 2.63-2.66 เป็นจำนวนลำดับของตัวกรองและ $0 \leq n \leq N - 1$ ดังนั้นถ้า $N = 7$ จะได้ $h(0) = h(6), h(1) = h(5), h(2) = h(4)$ และถ้า $N = 8$ จะได้ $h(0) = h(7), h(1) = h(6), h(2) = h(5), h(3) = h(4)$ และจากสมการที่ 2.63-2.66 สามารถนำมาเขียนเป็นกราฟได้ดังรูปที่ 2.27

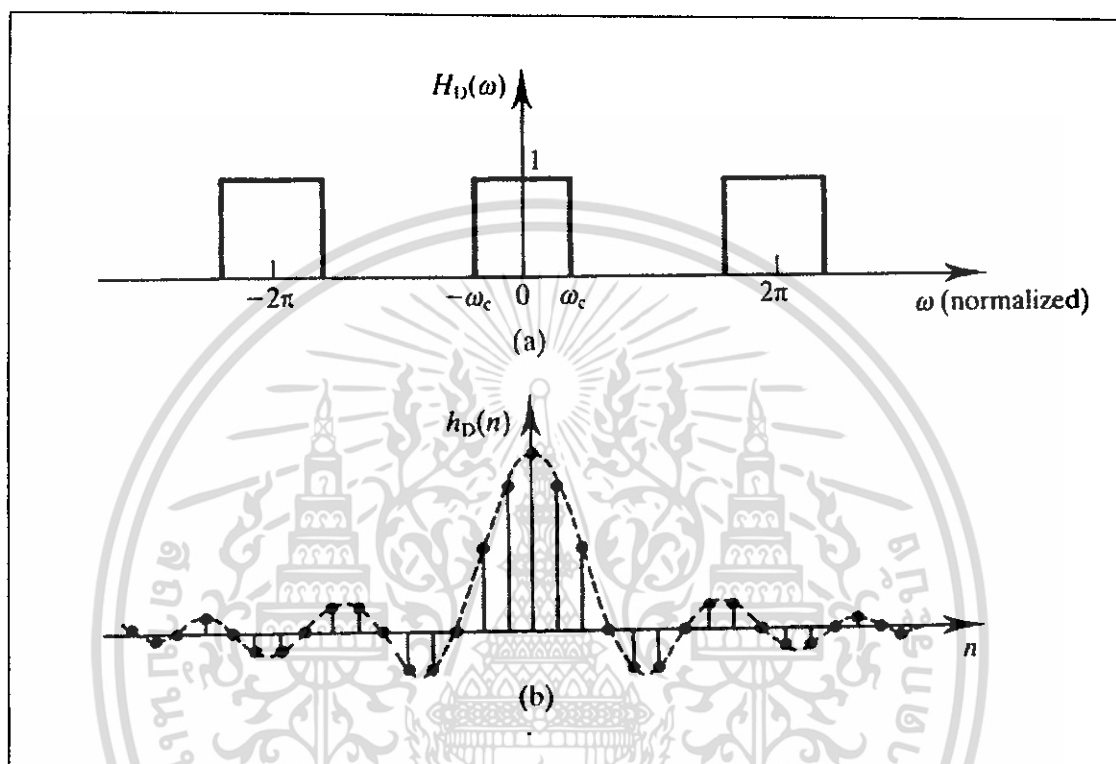


รูปที่ 2.27 แสดงผลตอบสนองอิมพัลส์ของเฟสเชิงเส้นของตัวกรอง 4 ชนิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4.1 การออกแบบตัวกรองดิจิทัลแบบ FIR

จะใช้วิธีฟูรีเยร์ซีรีส์ (Fourier Series) มาทำการออกแบบพิจารณารูปที่ 2.28



รูปที่ 2.28 แสดงผลตอบสนองความถี่และผลตอบสนองอิมพัลส์

(a) แสดงการตอบสนองความถี่ของตัวกรองแบบ Low-pass ในทางอุดมคติ

(b) แสดงผลตอบสนองอิมพัลส์ของตัวกรองแบบ Low-pass ในทางปฏิบัติ

ตัวกรองที่มีผลตอบสนองทางความถี่ $H_D(\omega)$ สามารถหาค่าผลตอบสนองอิมพัลส์ $h_D(n)$ ได้จากความสัมพันธ์ของการแปลงกลับของฟูรีเยร์ดังสมการที่ 2.67

$$h_D(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_D(\omega) e^{j\omega n} d\omega \quad (2.67)$$

พิจารณาจากรูปที่ 2.28 ซึ่งเป็นกราฟแสดงการตอบสนองความถี่ของตัวกรองแบบ Low-pass ซึ่งมีความถี่คัท-ออฟ คือ ω_c โดย ω_c นี้จะเป็นความถี่นอร์มอลไลซ์ (Normalized Frequency) ซึ่งมีค่าดังสมการที่ 2.68

$$f = \frac{f_c}{f_s} \quad (2.68)$$

โดย f คือ ความถี่คัท-ออฟนอร์มอลไลซ์ (Normalized Cut-off Frequency)

f_c คือ ความถี่คัท-ออฟ (Cut-off Frequency (Hz))

f_s คือ ความถี่สุ่ม (Sampling Frequency (Hz))

จากกราฟจะเห็นได้ว่าค่า $H_D(\omega)$ จะมีค่าเท่ากับ 1 ในช่วง ตั้งแต่ $-\omega_c$ จนถึง $+\omega_c$ ดังนั้นค่าของผลตอบสนองอิมพัลส์ของรูปที่ 2.28(a) จะหาได้จากสมการที่ 2.69

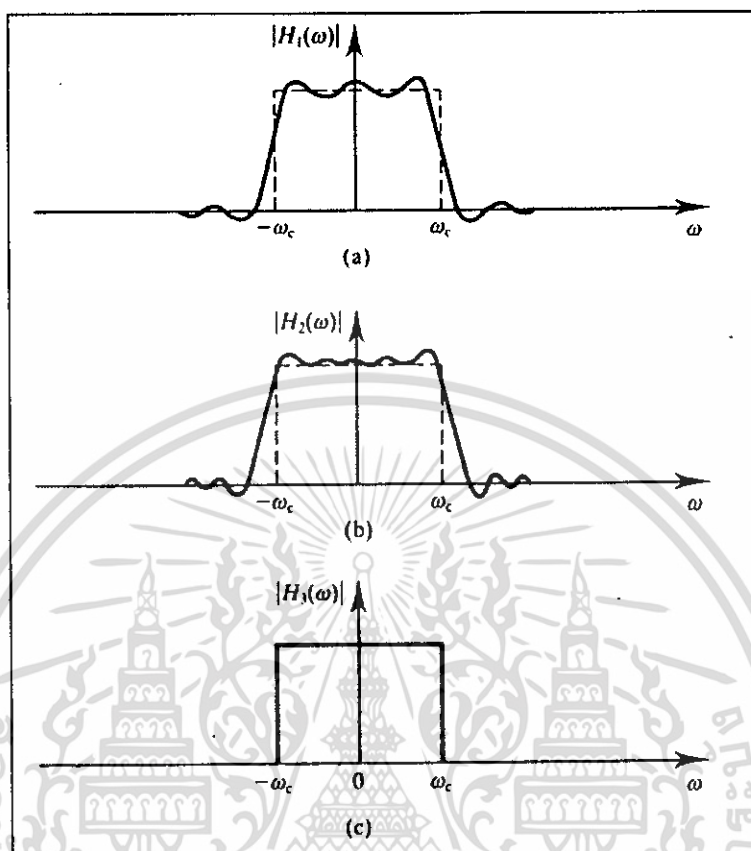
$$h_D(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} 1 \times e^{j\omega n} d\omega = \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} e^{j\omega n} d\omega \quad (2.69)$$

$$h_D(n) = \frac{2 \times \sin(2\pi fn)}{2\pi n} \quad (2.70)$$

$$h_D(n) = \frac{\sin(2\pi fn)}{\pi n} \quad (2.71)$$

เนื่องจากค่าผลตอบสนองอิมพัลส์นั้นมีความสมมาตรกันดังนั้นในการหาค่าจะทำการหาเพียงครึ่งหนึ่งก็พอ ดังตัวอย่าง ถ้าต้องการหาผลตอบสนองอิมพัลส์ $N = 53$ จะทำการหาค่า n ตั้งแต่ 0 จนถึง 26 ส่วนค่า -1 จนถึง -26 นั้นไม่จำเป็นต้องหาเนื่องจากความเป็นสมมาตรนั่นเอง

ตัวกรองเชิงเลขแบบ FIR จะมีผลตอบสนองทางความถี่เป็นแบบอุดมคติก็ต่อเมื่อมีการใช้ตัวสั้มประสิทธิ์เป็นอนันต์ แต่ในทางปฏิบัติผลการตอบสนองทางความถี่ของตัวกรองเชิงเลขแบบ FIR จะมีการกระเพื่อม (Ripple) เกิดขึ้น ถ้าจำนวนตัวสั้มประสิทธิ์มีค่าน้อยก็จะทำให้เกิดการกระเพื่อมมาก แต่ถ้าตัวสั้มประสิทธิ์มีจำนวนมากขึ้นก็จะทำให้ค่าการกระเพื่อมน้อยลงซึ่งจะมีผลใกล้เคียงกับผลการตอบสนองความถี่ทางอุดมคติมากขึ้น ดังแสดงในรูปที่ 2.29



รูปที่ 2.29 แสดงผลการตอบสนองความถี่

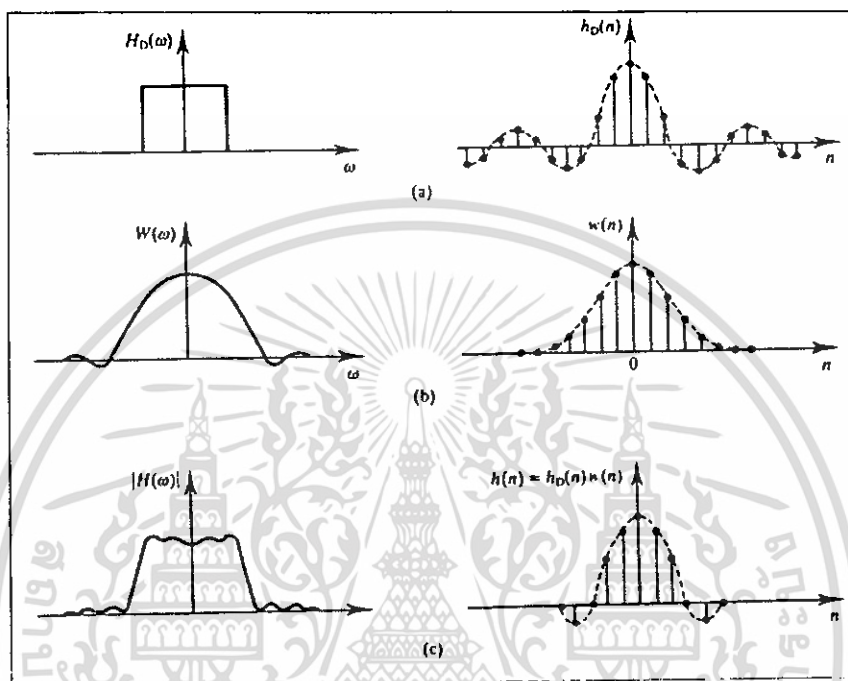
- (a) สัมประสิทธิ์ 13 ตัว
- (b) สัมประสิทธิ์ 25 ตัว
- (c) สัมประสิทธิ์เป็นอนันต์

2.2.4.2 การออกแบบตัวกรองดิจิทัลแบบ FIR ด้วยฟังก์ชันวินโดว์

เพื่อให้ผลการตอบสนองความถี่มีค่าใกล้เคียงกับผลตอบสนองความถี่ในทางอุดมคติจึงได้มีการค้นคว้าวิธีการแก้ปัญหาโดยการนำสัมประสิทธิ์ $h_D(n)$ มาคูณกับสัมประสิทธิ์อีกตัวหนึ่ง สัมประสิทธิ์ตัวนี้ก็คือ $\omega(n)$ หรือ Window Function ผลที่ได้จากการคูณจะเป็นสัมประสิทธิ์ตัวใหม่ หรือ $h(n)$ ซึ่งการหาสัมประสิทธิ์ $h(n)$ นี้เรียกว่า การหาโดยใช้ฟังก์ชันของวินโดว์ (Window) สมการของ $h(n)$ แสดงได้ดังสมการที่ 2.72

$$h(n) = h_D(n) \cdot \omega(n) \quad (2.72)$$

ค่าของ $\omega(n)$ สามารถหาได้หลายวิธีขึ้นอยู่กับความต้องการของผู้ออกแบบซึ่งแต่ละวิธีจะให้ผลการตอบสนองทางขนาดและผลตอบสนองทาง Impulse ที่แตกต่างกันดังแสดงในรูปที่ 2.30



รูปที่ 2.30 แสดงผลการตอบสนองความถี่ด้วยฟังก์ชันวินโดว์

นอกจากนี้สมการที่ใช้หาค่า $\omega(n)$ ของแต่ละวิธียังแตกต่างกันดังต่อไปนี้

- เรคแทนกูลาร์วินโดว์ (Rectangular Window)

$$\omega(n) = 1 \quad (2.73)$$

- ฮานนิงวินโดว์ (Hanning Window)

$$\omega(n) = 0.5 + 0.5 \cos\left(\frac{2\pi n}{N}\right) \quad (2.74)$$

- แฮมมิงวินโดว์ (Hamming Window)

$$\omega(n) = 0.54 + 0.46 \cos\left(\frac{2\pi n}{N}\right) \quad (2.75)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แบลคแมนวินโดว์ (Blackman Window)

$$\omega(n) = 0.42 + 0.5 \cos\left(\frac{2\pi n}{N-1}\right) + 0.08 \cos\left(\frac{4\pi n}{N-1}\right) \quad (2.76)$$

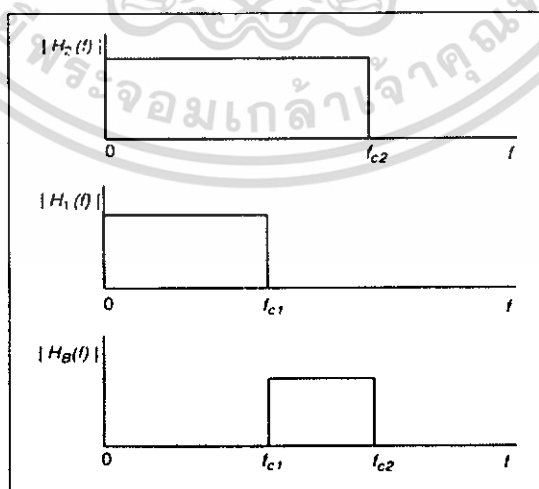
เมื่อ N คือ จำนวน Order

ฟังก์ชันวินโดว์	ความกว้าง Main Lobe	ความสูง Side Lobe, dB	Ripple ratio(%) ที่ $N=31$	ค่าการลดทอน ช่วงแถบหยุด, dB
Rectangular	$2\pi / N$	-13.5	21.8	-21
Hann	$4\pi / N$	-32	2.67	-44
Hamming	$4\pi / N$	-43	0.82	-53
Blackman	$6\pi / N$	-57	0.12	-74

ตารางที่ 2.4 เปรียบเทียบฟังก์ชันหน้าต่าง

2.2.4.3 การออกแบบตัวกรองช่วงความถี่ผ่านแบบ FIR

ตัวกรองช่วงความถี่ผ่าน (Band-pass Filter) ในการออกแบบตัวกรองช่วงความถี่ผ่านแบบ FIR จะทำโดยนำเอาผลการตอบสนองความถี่ของตัวกรองแบบ Low-pass 2 ความถี่มาลบกันดังแสดงในรูปที่ 2.31



รูปที่ 2.31 แสดงกราฟผลตอบสนองความถี่ของตัวกรองแบบ Low-pass 2 ความถี่ และกราฟผลตอบสนองความถี่ของตัวกรองแบบ Band-pass

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และในการหาค่าสัมประสิทธิ์ของตัวกรองช่วงความถี่ผ่านสามารถหาได้จากสมการ

$$h_D(n)_{Band} = h_D(n)_{fc2} - h_D(n)_{fc1} \quad (2.77)$$

$$h_D(n)_{Band} = \frac{\sin(2\pi f_{c2}n)}{n\pi} - \frac{\sin(2\pi f_{c1}n)}{n\pi} \quad (2.78)$$

2.2.5 FPGA (Field Programmable Gate Array)

ในปัจจุบัน FPGA ได้มีการใช้งานกันอย่างกว้างขวางมากขึ้น โดย FPGA ได้ถูกนำมาแทนที่ Logic Gate ทั้งในระดับ SSI MSI หรือแม้แต่ในส่วนประมวลผลก็ตาม โดยจะเห็นได้จากอุปกรณ์ที่เป็นส่วนประกอบของระบบคอมพิวเตอร์ ทั้งนี้เนื่องจาก FPGA มีข้อดีอยู่หลายประการ ดังต่อไปนี้

- สามารถออกแบบวงจรดิจิทัลได้โดยมีจำนวนอุปกรณ์และขนาดของแผงวงจรลดลง เนื่องจากวงจรถูกออกแบบไว้ในชิพ
- วงจรที่ออกแบบสามารถทำงานที่ความถี่สูงกว่าไมโครคอนโทรลเลอร์ เพราะ FPGA มีลักษณะการทำงานเป็นแบบฮาร์ดแวร์
- สามารถแก้ไขการทำงานได้ง่าย เพราะเนื่องจากลักษณะการทำงานแบบฮาร์ดแวร์ จะถูกกำหนดและออกแบบโดยภาษาระดับสูง (High Level Design)

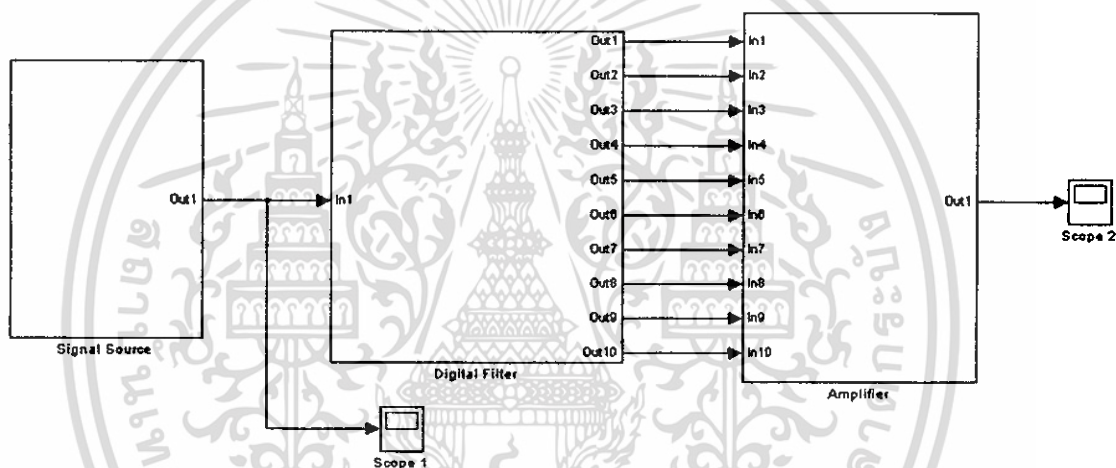
บทที่ 3

การทดลอง (ทฤษฎี)

ได้ทำการสร้าง โมเดลขึ้นจากฟังก์ชัน Simulink ของโปรแกรม Matlab

3.1 การทดลองที่ 1 ทดสอบอ็ควอไลเซอร์ที่มีลำดับตัวกรองต่ำ (53) กับสัญญาณชานัน

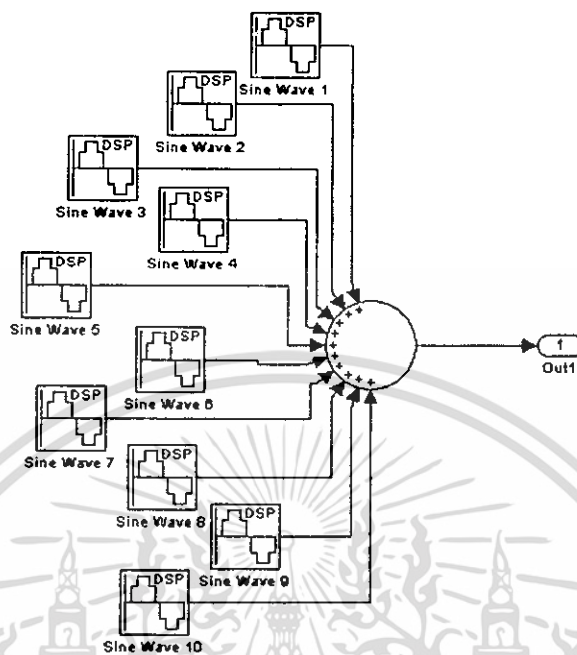
เป็นการทดลองจำลองการทำงานของอ็ควอไลเซอร์กับสัญญาณชานันที่มีค่าความถี่และแอมพลิจูดที่แตกต่างกันไป



รูปที่ 3.1 โมเดลของอ็ควอไลเซอร์ที่ใช้อินพุตเป็นสัญญาณชานัน

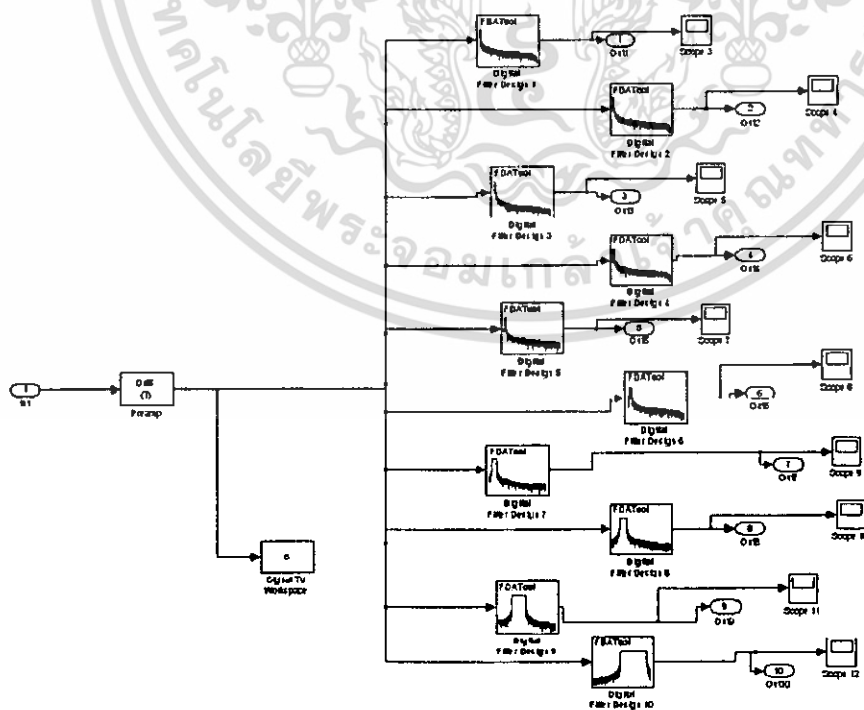
รูปที่ 3.1 โมเดลประกอบไปด้วย

- ส่วนสร้างสัญญาณความถี่และขนาดต่างๆประกอบด้วยบล็อกสัญญาณชานันที่ความถี่และขนาดต่างๆจากนั้นรวมสัญญาณเข้าเป็นสัญญาณอินพุต



รูปที่ 3.2 โมเดลสร้างสัญญาณซายน์

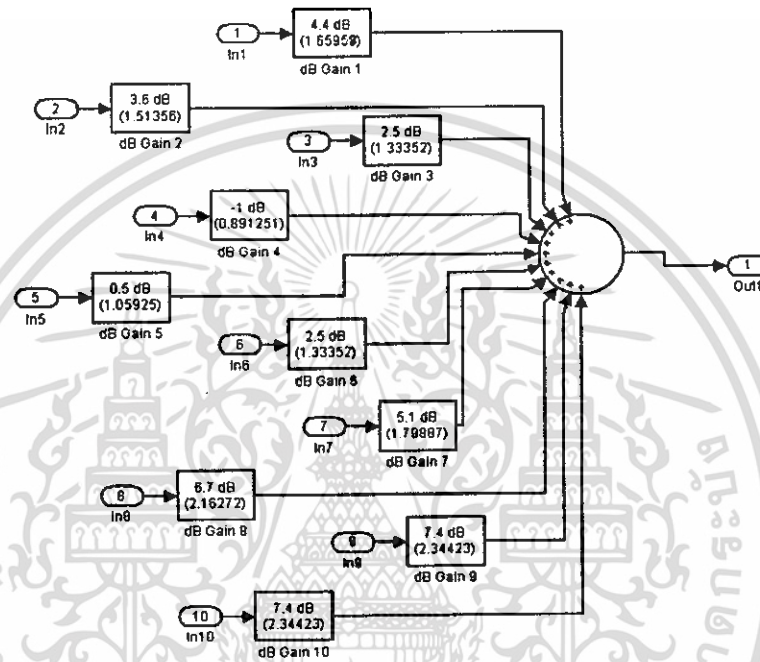
- ส่วนบล็อควงจรกรองสัญญาณเชิงเลขประกอบด้วยส่วนขยายภาคคั่นและบล็อควงจรกรองความถี่เชิงเลข



รูปที่ 3.3 ตัวกรองความถี่ทั้ง 10 ช่องสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วนวงจรขยายสัญญาณกำหนดอัตราขยายที่ความถี่ช่วงต่างๆ โดยมีหน่วยเป็น เดซิเบล เมื่อขยายแล้วจึงทำการรวมสัญญาณของช่วงความถี่ต่างๆ โดยในการทดลองทั้งหมดให้ตัวขยายของทุกช่วงความถี่เป็น 0 dB (ไม่ขยาย)



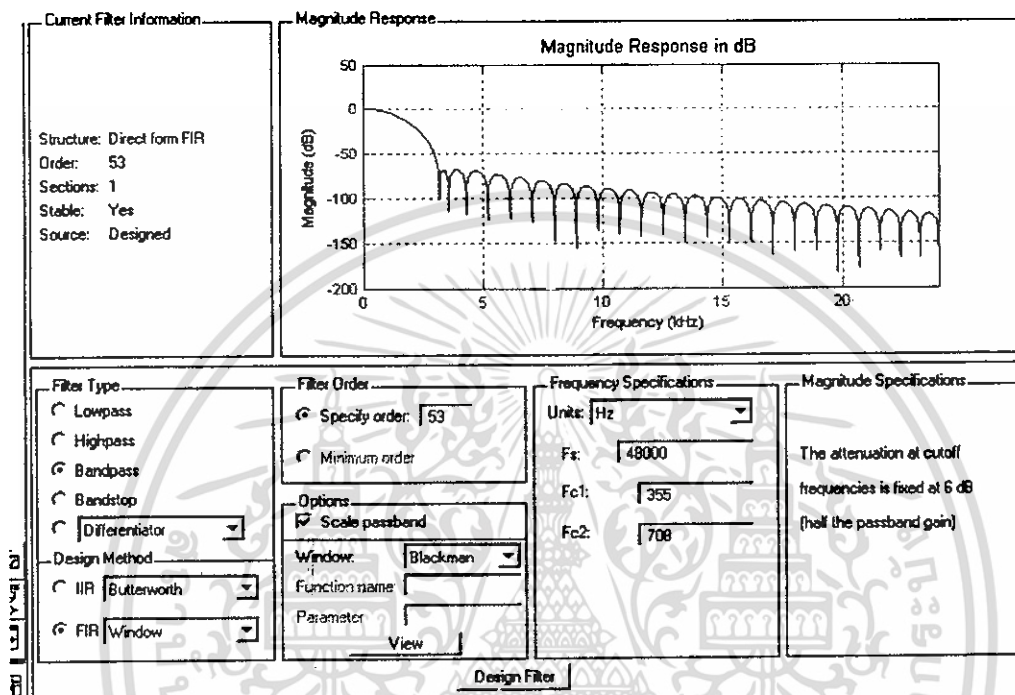
รูปที่ 3.4 ส่วนขยายสัญญาณ

- ส่วนจับภาพสัญญาณ

วิธีการจำลองการทำงานของโมเดล

1. กำหนดความถี่ 10 แหล่งกำเนิดให้มีค่าเป็นความถี่กลางของวงจรกรองความถี่แถบผ่านทั้ง 10 ช่วง โดยมีขนาดแอมพลิจูดตั้งแต่ 1 ถึง 10
2. กำหนดให้ เวลาสุ่มของแหล่งจ่าย มีค่าเท่ากับ $1/48000$ s เท่ากับความถี่สุ่มของตัวกรองความถี่
3. เปิดบล็อกเซตของวงจรกรองความถี่เชิงเลข ใช้วงจรกรองความถี่เชิงเลขแถบผ่านแบบไม่ป้อนกลับ (FIR) และใช้หน้าต่างแบบสี่เหลี่ยม กำหนดช่วงความถี่แถบผ่านต่างๆ และให้ความถี่สุ่มเท่ากับ 48KHz

4. กำหนดอันดับของวงจรให้มีค่าเท่ากับ 53 กดที่ปุ่ม design filter จะเป็นการแสดงกราฟการตอบสนองความถี่ดังภาพ 3.5



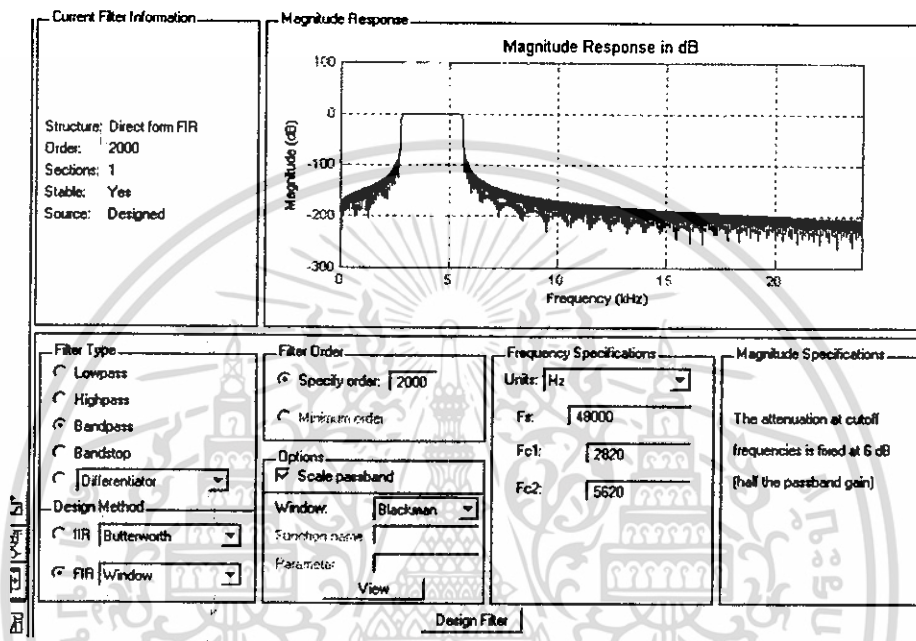
รูปที่ 3.5 บล็อกตัวกรองสัญญาณที่มีตัวกรองลำดับต่ำ (53)

5. เมื่อกำหนดค่าเรียบร้อยแล้วให้กำหนดเวลาที่ต้องการประมวลผล จากนั้น กด F5 เพื่อทำการจำลองการทำงาน

6. สามารถดูภาพสัญญาณอินพุตได้ที่ Scope 1 สัญญาณเอาต์พุตที่ Scope 2 และสัญญาณที่ออกมาจากตัวกรองความถี่เชิงเลขแต่ละจุดที่ Scope 3 - 12

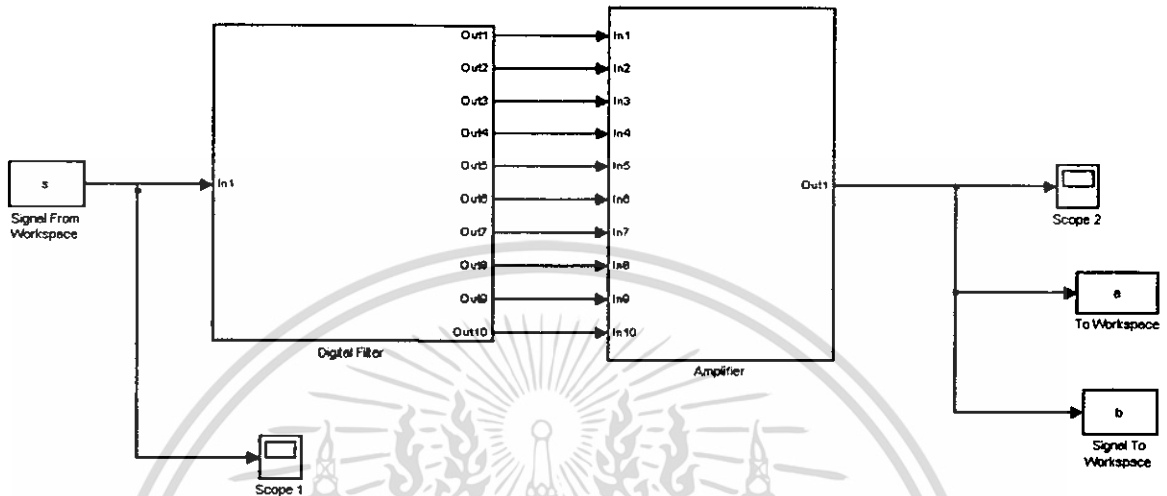
3.2 การทดลองที่ 2 ทดสอบอีควอไลเซอร์ที่มีลำดับของตัวกรองสูง (2000) กับสัญญาณซายน์

- จากโมเดลที่แล้วให้กำหนดค่าเหมือนเดิมยกเว้นอันดับของวงจรกรองความถี่เชิงเลขให้เปลี่ยนเป็น 2000 ให้สังเกตถึงรูปสัญญาณในแต่ละจุดที่เปลี่ยนไป



รูปที่ 3.6 บล็อกตัวกรองสัญญาณที่มีตัวกรองลำดับสูง (2000)

3.3 การทดลองที่ 3 ทดสอบอ็ควอไลเซอร์ที่มีลำดับตัวกรองสูง (2000) กับ ไฟล์เสียง



รูปที่ 3.7 โมเดลของอ็ควอไลเซอร์ที่ใช้อินพุตเป็นไฟล์เสียงจริง

รูปที่ 3.7 โมเดลประกอบไปด้วย

- ส่วนรับค่าอินพุตที่เป็นไฟล์เสียงจาก Workspace
- ส่วนบล็อกตัวกรองความถี่เชิงเลข
- ส่วนบล็อกขยายสัญญาณ

โดยแตกต่างกับการทดลองกับที่แล้วว่าแหล่งกำเนิดสัญญาณและบล็อกวงจรกรองความถี่เชิงเลขที่จะรับสัญญาณเสียงเข้ามาจากไฟล์ .wav ภายนอกเข้ามา

วิธีการจำลองการทำงานของโมเดล

1. โหลดไฟล์เสียงเข้าไปที่ Workspace ของโปรแกรม Matlab โดยใช้คำสั่ง
`[s,fs]=wavread('directoryของไฟล์');`

โปรแกรมจะทำการ โหลดไฟล์เสียงเข้ามาเก็บไว้ที่ workspace ที่ตัวแปร s

2. เปิดบล็อกเซตของวงจรกรองความถี่เชิงเลขใช้วงจรกรองความถี่แถบผ่าน และแบบหน้าต่างแบล็กแมน กำหนดช่วงความถี่ต่างๆและให้ความถี่สุ่มเท่ากับ 48KHz

3. กำหนดอันดับของวงจรให้มีค่าเท่ากับ 2000 กคที่ปุ่ม design filter จะเป็นการแสดงกราฟการตอบสนองความถี่เหมือนรูปที่ 3.6

4. เมื่อกำหนดค่าเรียบร้อยแล้วให้กำหนดเวลาที่ต้องการประมวลผล จากนั้น กด F5 เพื่อทำการจำลองการทำงาน โปรแกรมส่งค่าสัญญาณเอาต์พุตกลับไป Workspace ในค่าของตัวแปร a และ b

5. โปรแกรม Matlab สามารถเล่นไฟล์เสียงที่เป็นอาร์เรย์ 2 มิติเท่านั้นแต่ไฟล์ที่ส่งกลับมาเป็น อาร์เรย์ 3 มิติ จึงต้องใช้คำสั่ง

`X=squeeze(b);` หรือ

`Y=squeeze(a);`

ทำการตัดมิติที่มีค่าเท่ากับ 1 ออกและใช้คำสั่งเปิด ไฟล์เสียง

`Sound(x,48000);`

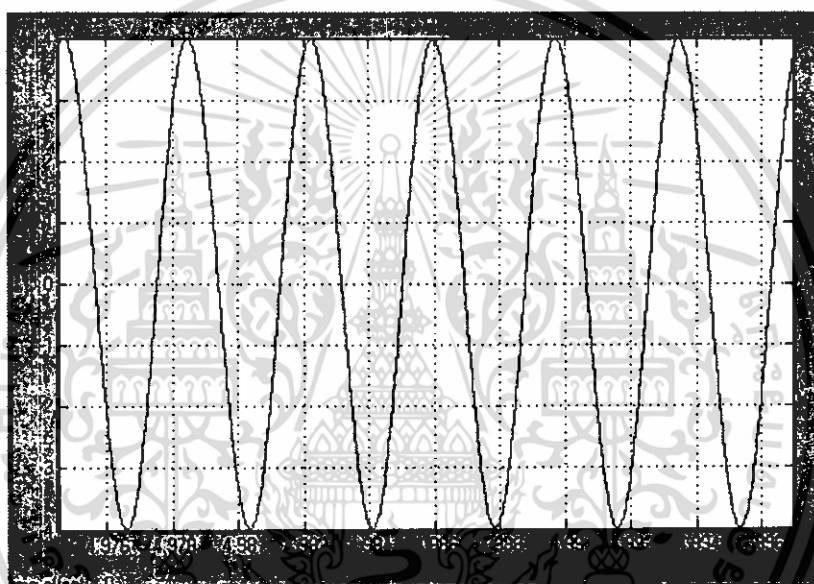
เสียงที่ออกมาจะมีการเปลี่ยนแปลงตามค่าอันดับของวงจรรองความถี่และอัตราขยายที่กำหนดไว้ยังมีอันดับมากคุณภาพเสียงจะยิ่งดีเนื่องจากมีผลการตอบสนองทางความถี่ของวงจรดีขึ้น แต่ส่งผลให้ต้องเวลาในการจำลองการทำงานมากขึ้นด้วย



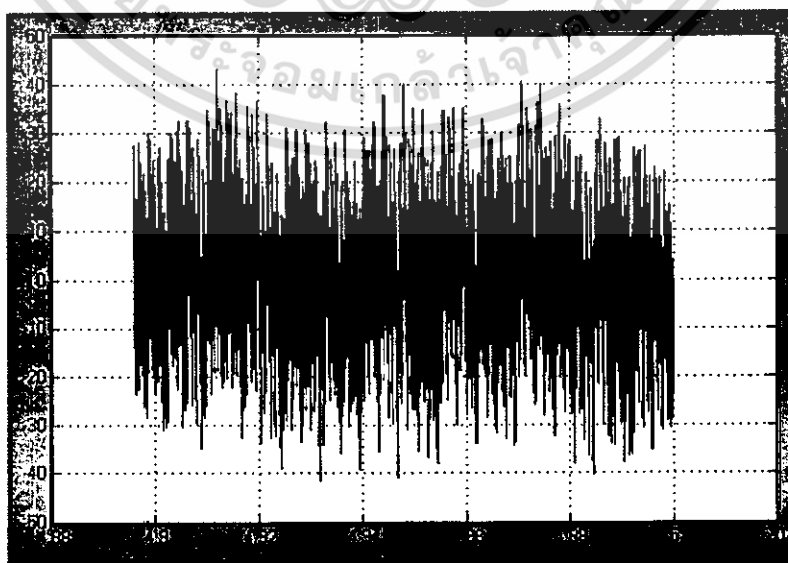
บทที่ 4

ผลการทดลอง (ทฤษฎี)

4.1 ผลการทดลอง อีควอไลเซอร์ที่มีลำดับของตัวกรองต่ำ (53) กับสัญญาณชาชนัน ตัวอย่างสัญญาณอินพุตชาชนัน



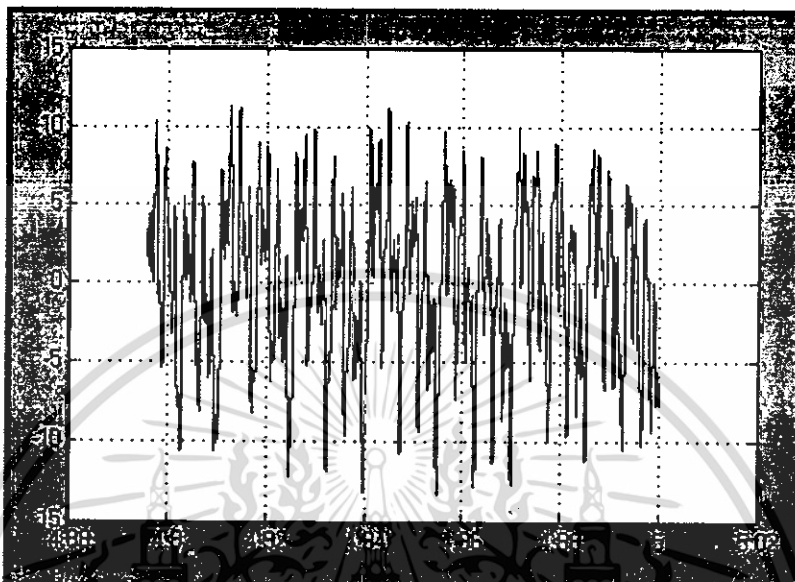
รูปที่ 4.1 สัญญาณชาชนันอินพุต



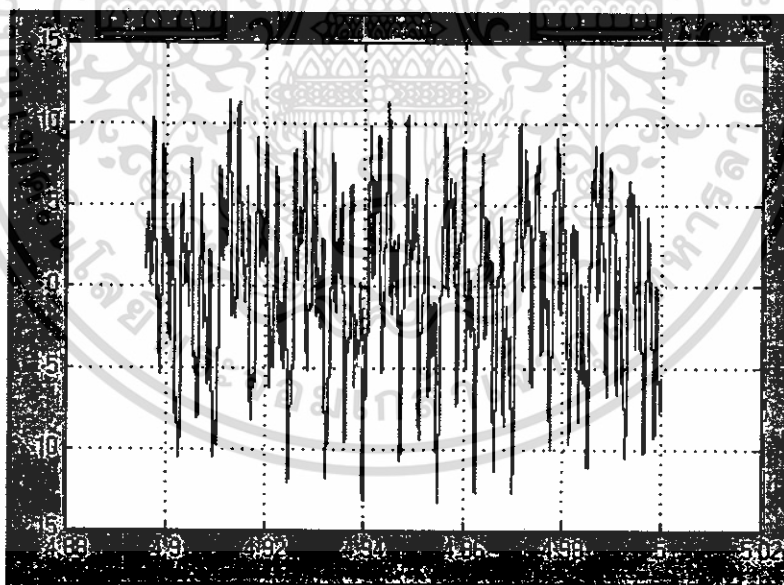
รูปที่ 4.2 สัญญาณรวมชาชนันอินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณเอชทีพูทหลังจากผ่านตัวกรองที่มีความถี่ต่างๆ

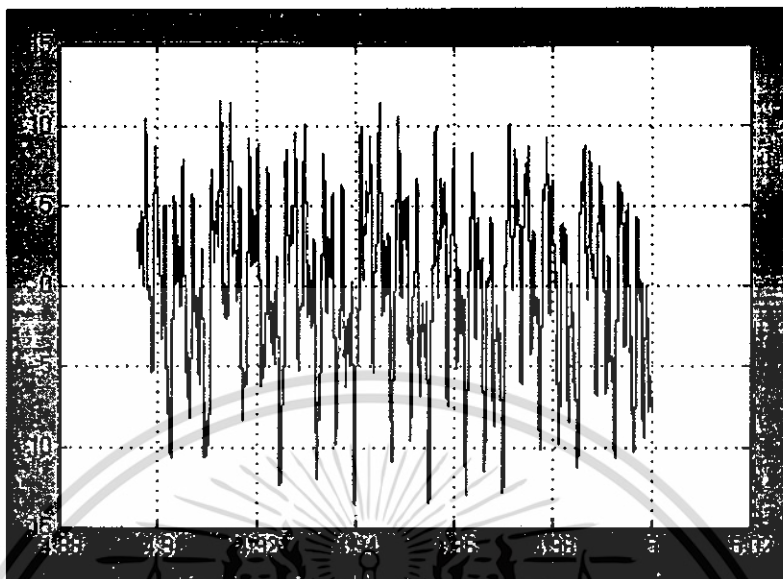


รูปที่ 4.3 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 33Hz

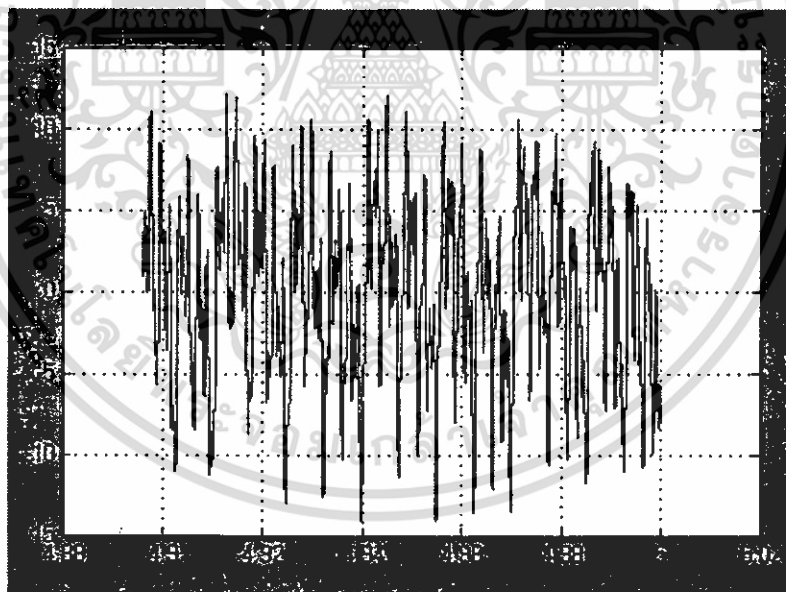


รูปที่ 4.4 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 67Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

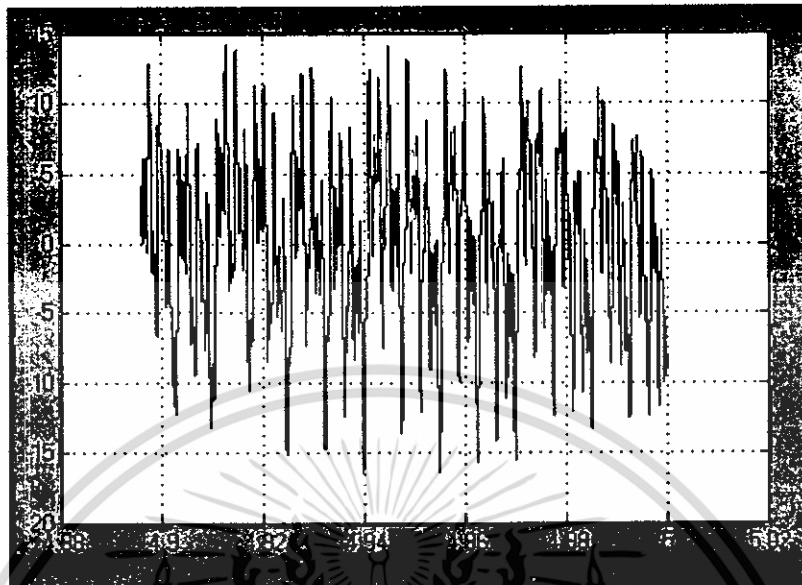


รูปที่ 4.5 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 134Hz

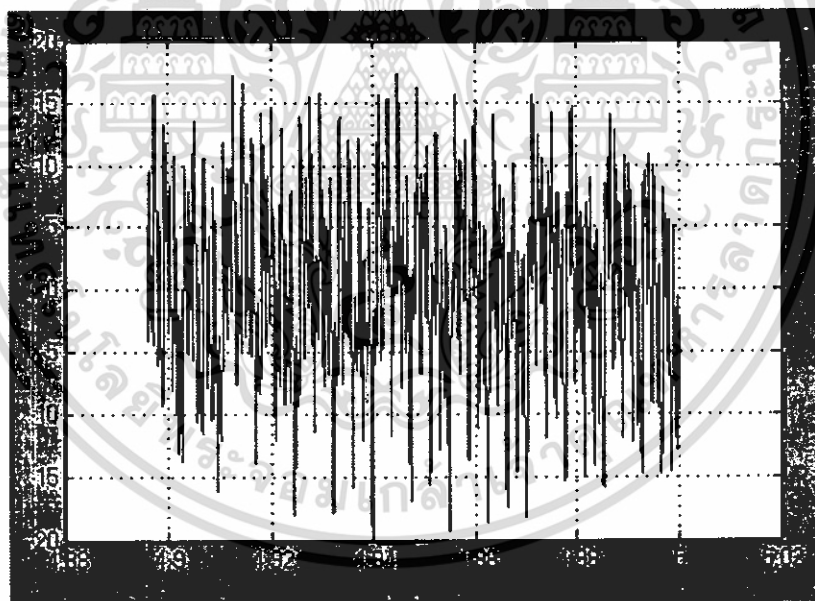


รูปที่ 4.6 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 267Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

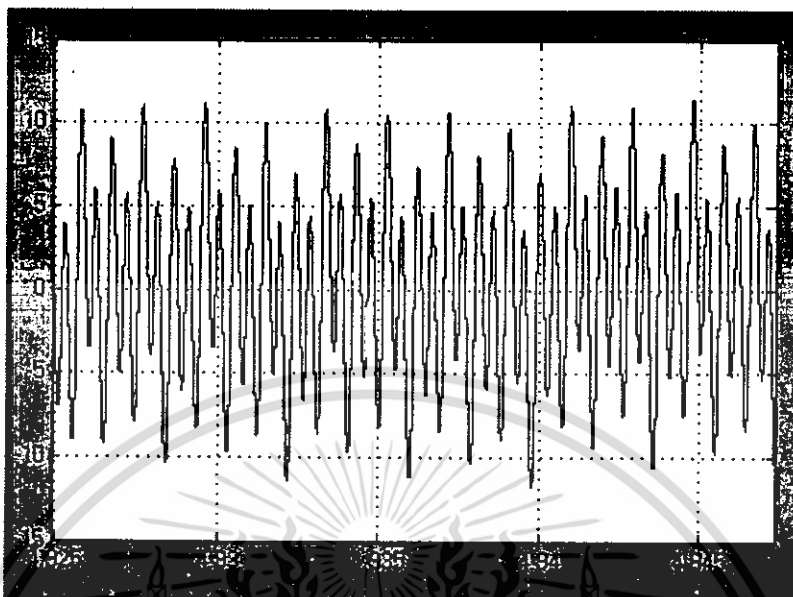


รูปที่ 4.7 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 532Hz

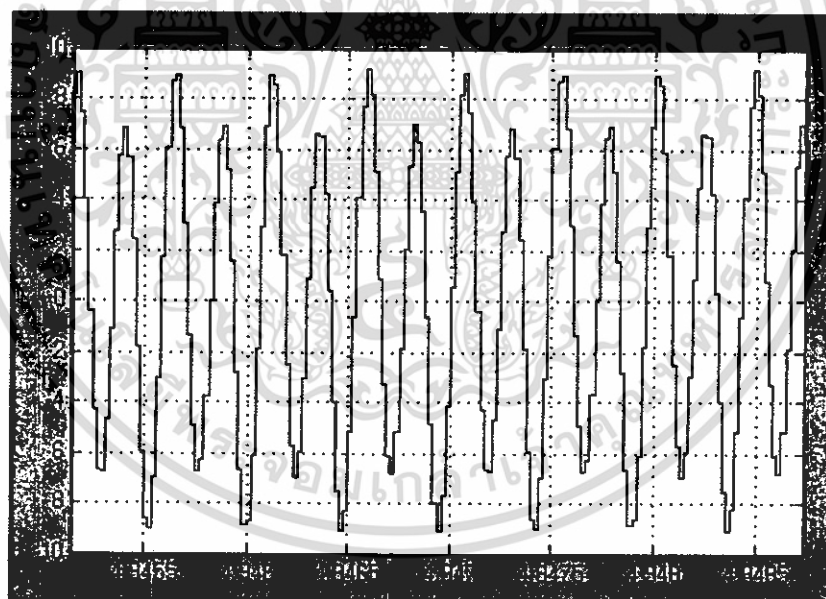


รูปที่ 4.8 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 1055Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

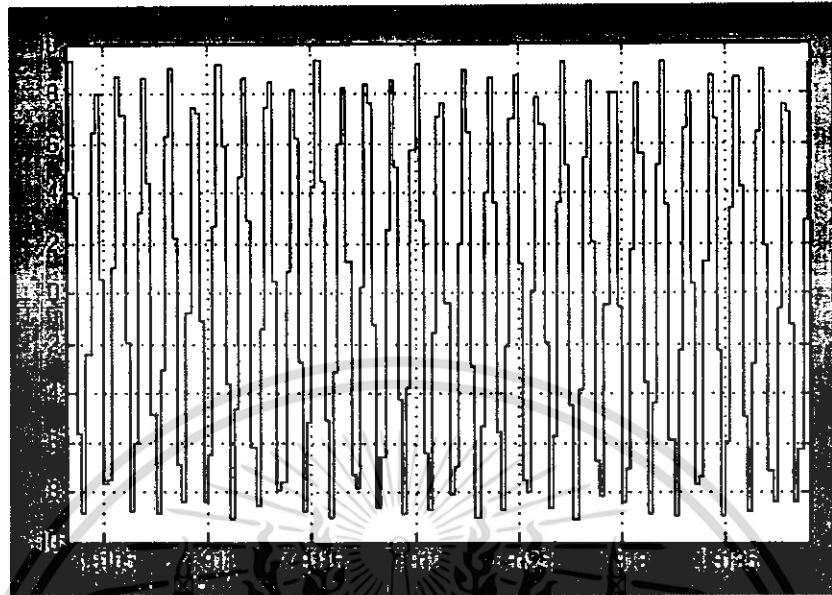


รูปที่ 4.9 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 2110Hz

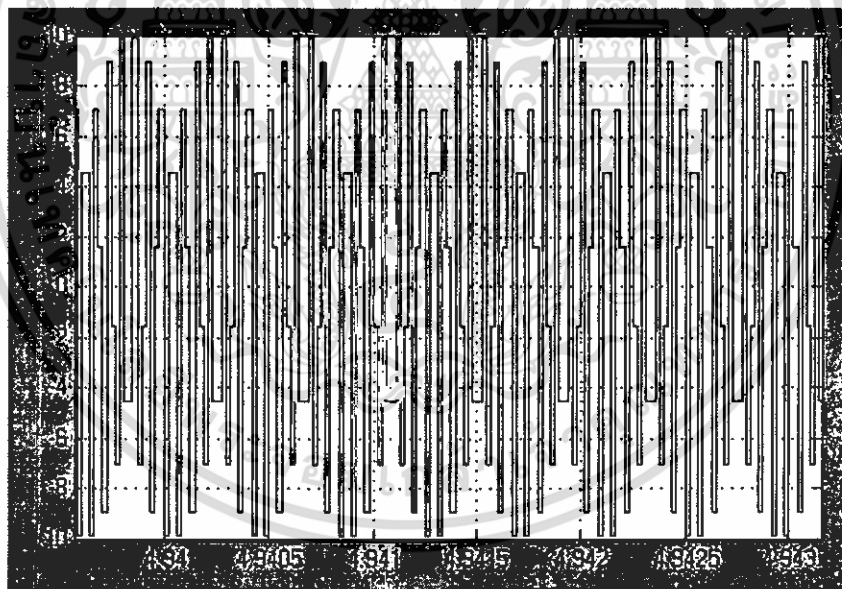


รูปที่ 4.10 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 4220Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



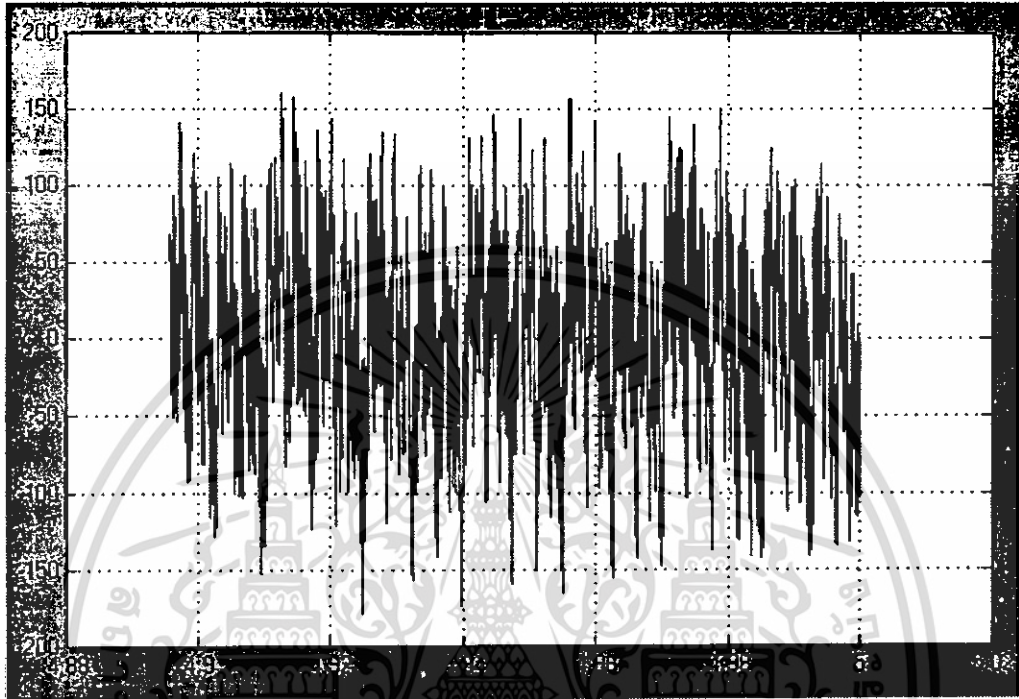
รูปที่ 4.11 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 8410Hz



รูปที่ 4.12 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 16800Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเมื่อรวมสัญญาณเอาร์ทพุทเข้าด้วยกัน

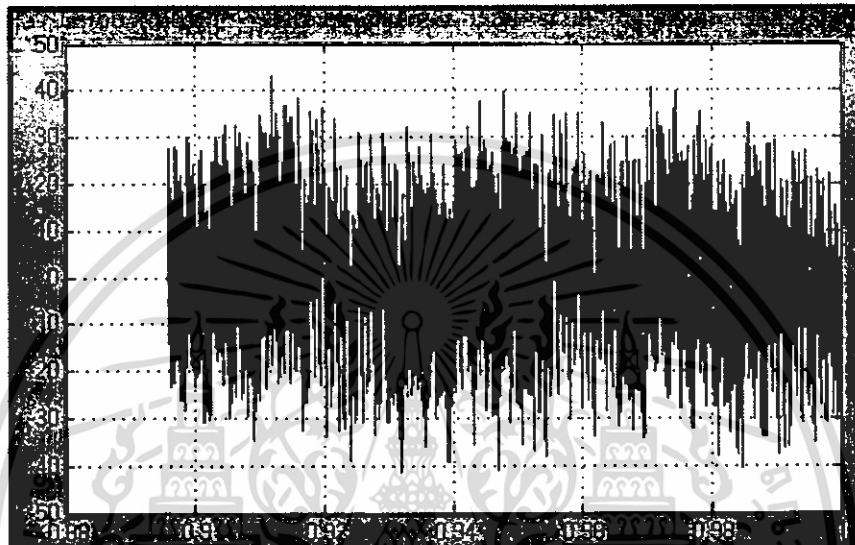


รูปที่ 4.13 สัญญาณรวมของสัญญาณเอาร์ทพุท

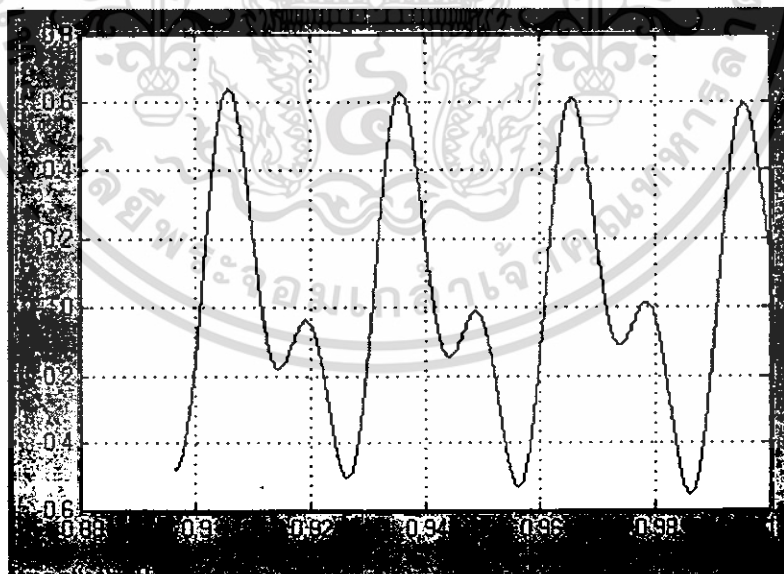
จากรูปที่ 4.13 จะเห็นได้ว่าสัญญาณซายน์ที่รวมเอาร์ทพุทมีแอมพลิจูด มากกว่า สัญญาณซายน์รวมที่อินพุทเสียอีก ทั้งๆที่ ส่วนขยายสัญญาณ ไม่ได้ขยายแน่แท้แค่หนึ่งเดียว จึงกล่าวได้ว่าสัญญาณรวมที่เอาร์ทพุทนี้มีความผิดพลาดอยู่สูงมาก

4.2 ผลการทดลอง อีควอไลเซอร์ที่มีลำดับตัวกรองสูง (2000) กับสัญญาณชายน์

สัญญาณเอาต์พุตหลังจากผ่านตัวกรองที่ความถี่ต่างๆ

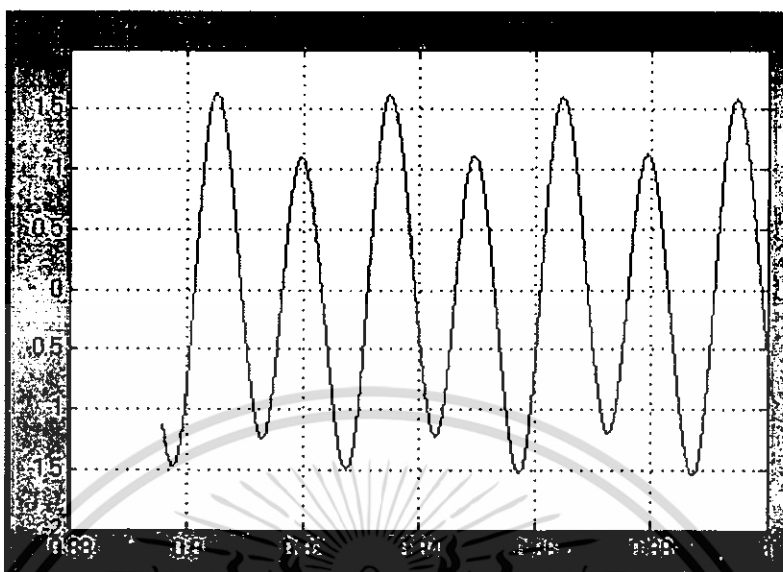


รูปที่ 4.14 สัญญาณรวมชายน์อินพุต

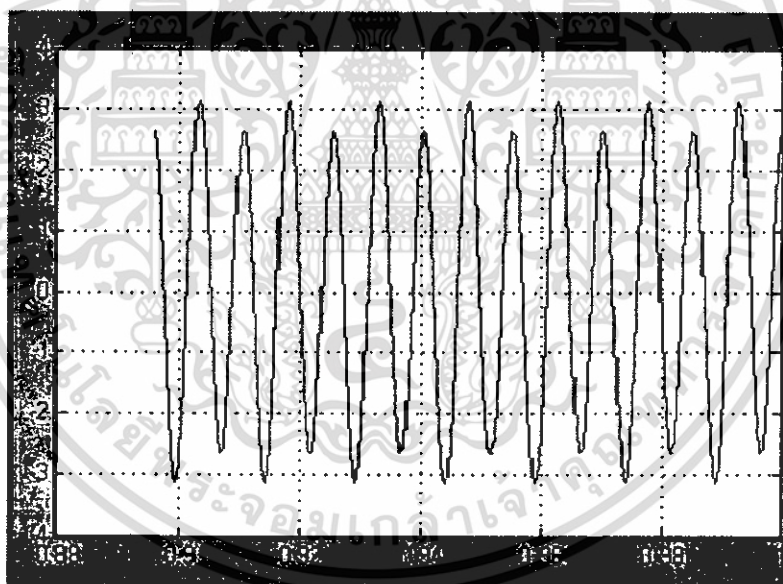


รูปที่ 4.15 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 33Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

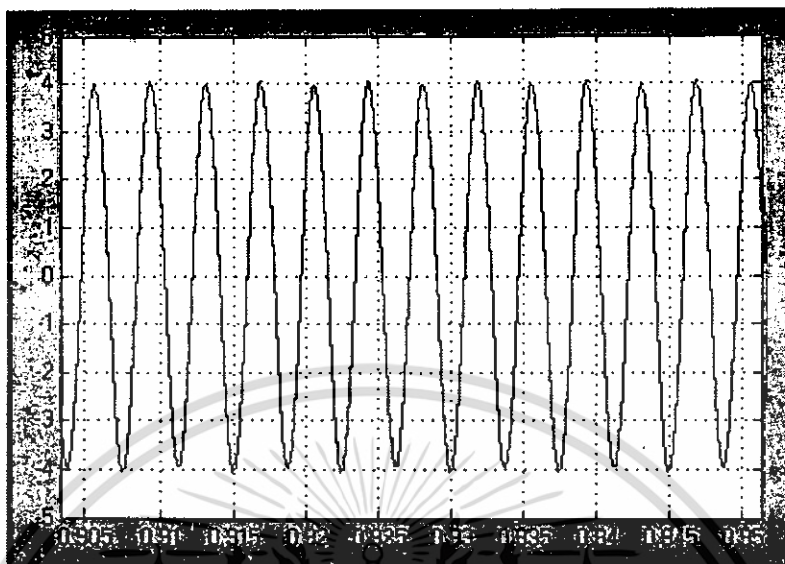


รูปที่ 4.16 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 67Hz

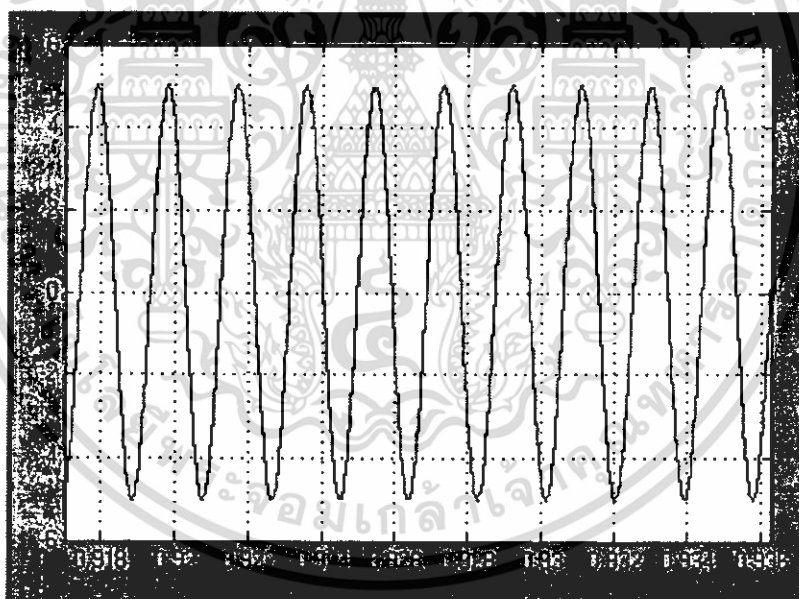


รูปที่ 4.17 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 134Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

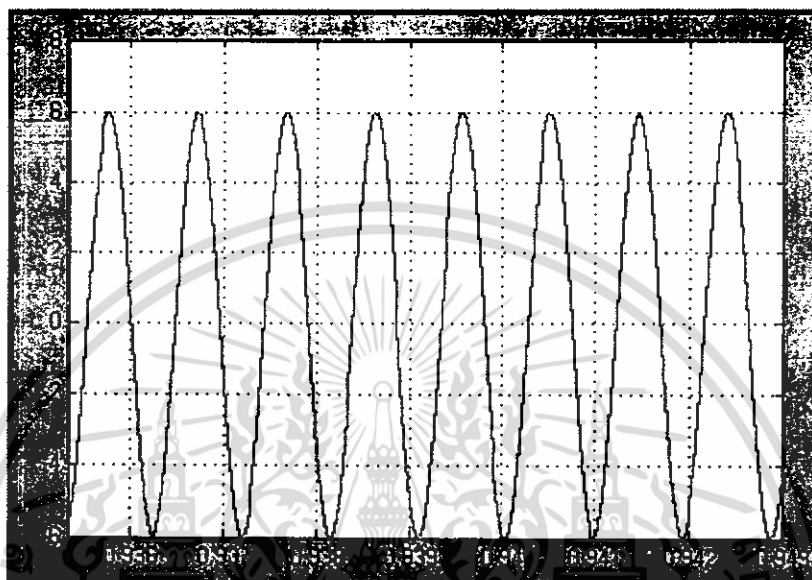


รูปที่ 4.18 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 267Hz

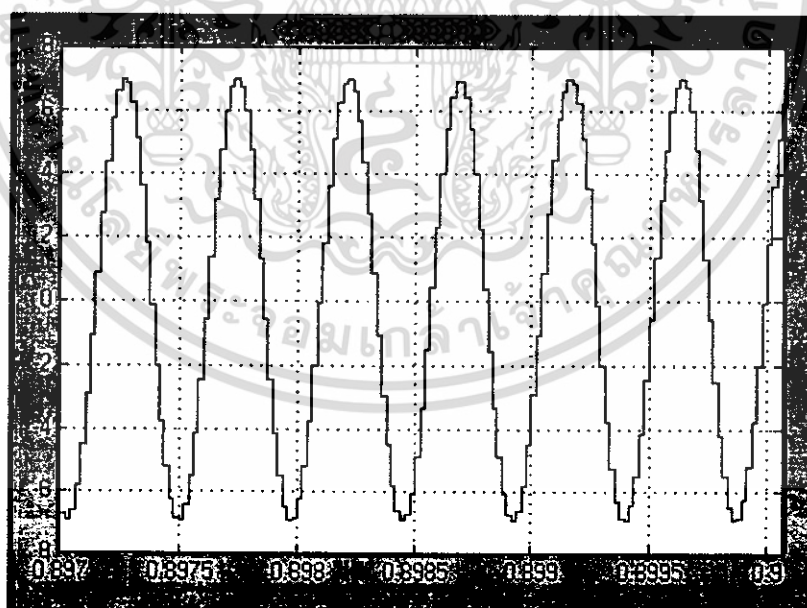


รูปที่ 4.19 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 532Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

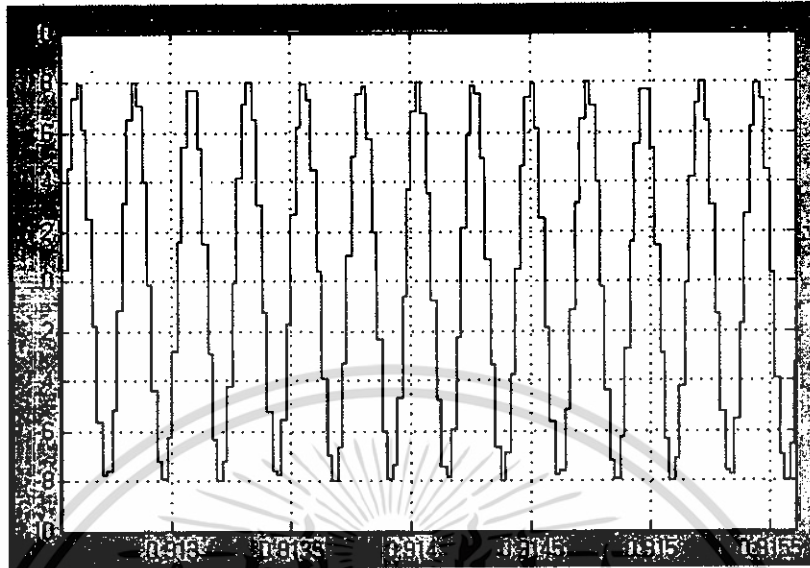


รูปที่ 4.20 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 1055Hz

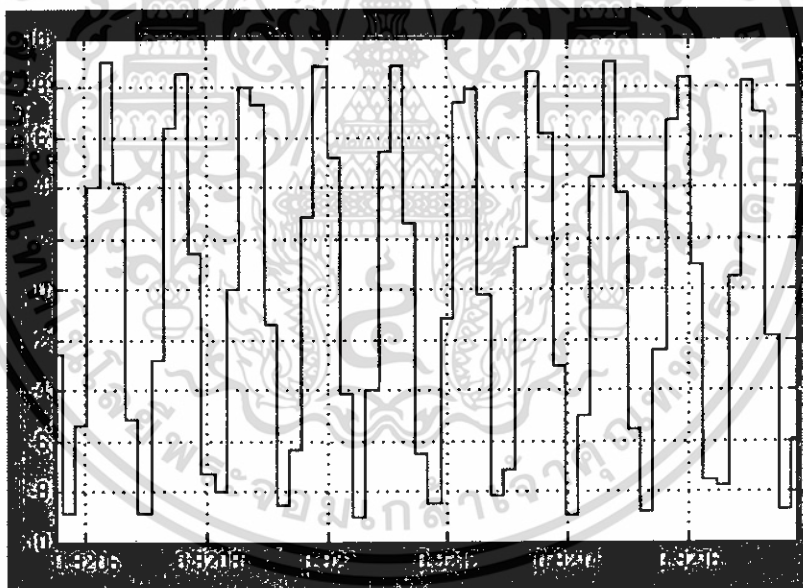


รูปที่ 4.21 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 2110Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

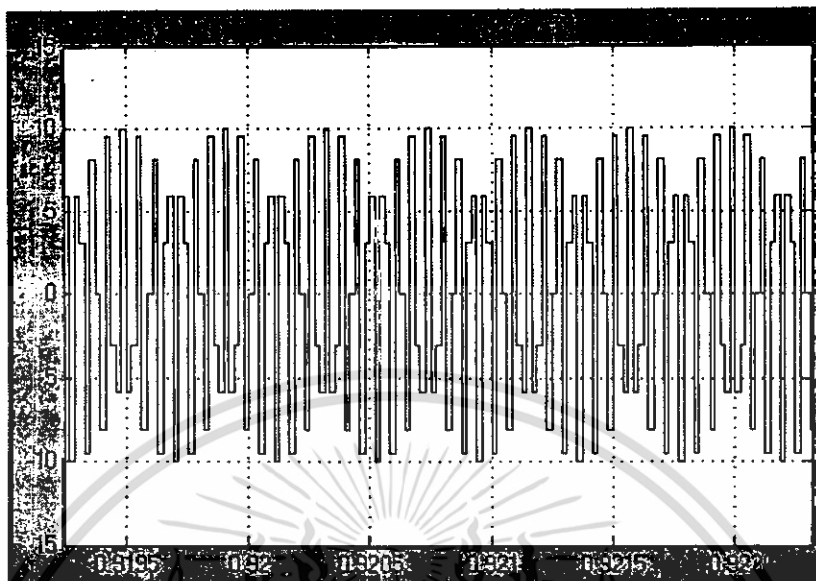


รูปที่ 4.22 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 4220Hz

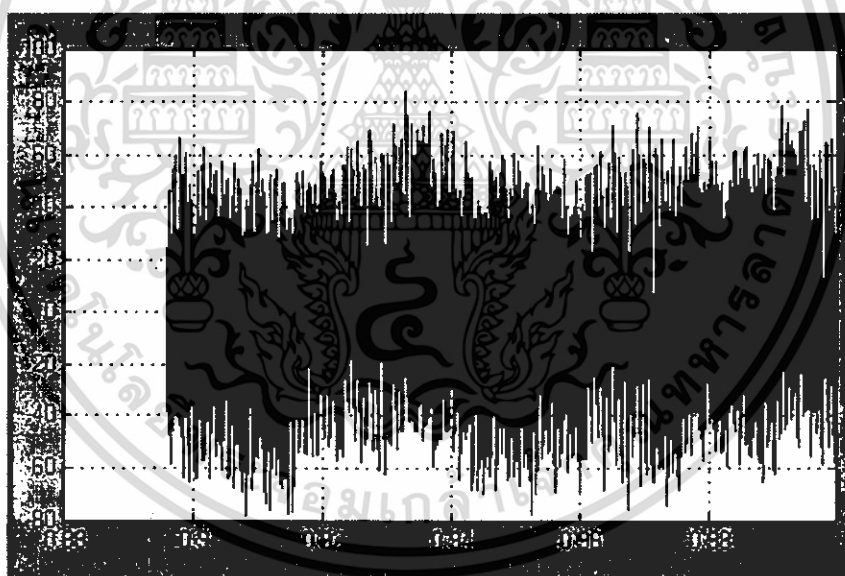


รูปที่ 4.23 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 8410Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.24 สัญญาณหลังจากผ่านตัวกรองที่มีความถี่กลางที่ 16800Hz



รูปที่ 4.25 สัญญาณรวมของสัญญาณเอาท์พุท

เมื่อเปรียบเทียบรูปที่ 4.13 กับรูปที่ 4.25 จะพบว่า รูปที่ 4.25 มีระดับของสัญญาณต่ำกว่ามาก เนื่องจากตัวกรองที่มีลำดับสูง ทำให้สามารถกรองความถี่ได้ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลอง (ทฤษฎี)

5.1 สรุปผลการทดลอง

จากการทดลองในขั้นต้นโดยการสร้างอีควอไลเซอร์ใน MATLAB พบว่า

- ทดลองสร้างอีควอไลเซอร์ โดยให้ตัวกรองมีลำดับน้อยๆ (53) จากนั้นป้อนสัญญาณขาเข้าที่มีความถี่ต่างๆ เข้าไป พบว่าสามารถกรองสัญญาณได้ดี แต่เมื่อทดลองป้อนสัญญาณเสียงพูดจริง ผลลัพธ์ที่ได้มีความเพี้ยนสูงมาก เนื่องจากเสียงพูดจริงประกอบไปด้วยความถี่มากมาย ทำให้ตัวกรองที่ไม่ใช่ตัวกรองในอุดมคติ ไม่สามารถกรองความถี่ได้หมด สัญญาณต่างๆ จึงมารวมกันทำให้เกิดความเพี้ยนสูง

- ทดลองสร้างอีควอไลเซอร์ โดยให้ตัวกรองมีลำดับหลายๆ (2000) ป้อนสัญญาณเสียงพูดจริง พบว่ากรองได้ดี โดยสามารถเน้นความถี่ช่วงหนึ่งๆ ได้อย่างชัดเจน และการป้อนสัญญาณขาเข้าก็ทำการกรองได้ดี

เมื่อเปรียบเทียบผลการทดลองทั้งหมดแล้ว พบว่ายิ่งตัวกรองมีลำดับมากก็จะสามารถกรองความถี่ได้ดียิ่งขึ้นซึ่งตรงกับทฤษฎี

จากการทดลองพบว่าการสร้างตัวกรองเชิงเลขโดยใช้หน้าต่างพื้นฐานต้องใช้ตัวกรองลำดับสูงมากจึงจะทำงานได้ดี แต่ตัวกรองอันดับสูงมากนั้นสร้างได้ยากและมีราคาแพง

5.2 ปัญหาที่พบ

- การใช้ FPGA ในการสร้างตัวกรองเชิงเลขแบบ FIR ยิ่งตัวกรองมีลำดับสูงก็จะยิ่งใช้ทรัพยากรภายในชิพมาก จึงทำให้การสร้างตัวกรองที่มีลำดับหลายๆ จำเป็นต้องใช้ชิพ FPGA ที่มีจำนวนเกตมากขึ้นตามไปด้วย

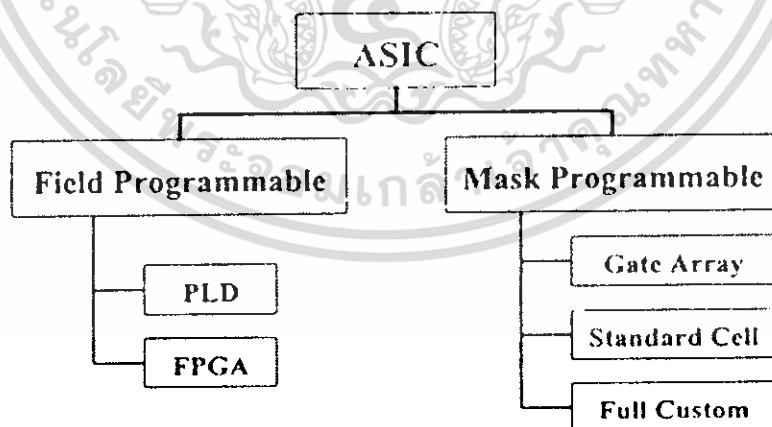
5.3 แนวทางการแก้ปัญหา

- เนื่องจาก FPGA ที่มีทรัพยากรภายในมากๆ มีราคาแพงแนวทางการแก้ปัญหาคือ ลดการใช้ทรัพยากรในส่วนอื่นๆ ลงศึกษาหาตัวกรองที่มีประสิทธิภาพและหาเทคนิคต่างๆ เพื่อช่วยลดการใช้ทรัพยากรลง

บทที่ 6

สถาปัตยกรรมและการออกแบบบน FPGA

ปัจจุบันอุตสาหกรรมทางด้านอิเล็กทรอนิกส์มีความรวดเร็วและอีกทั้งประสิทธิภาพที่เพิ่มขึ้นของอุปกรณ์อิเล็กทรอนิกส์ เพื่อให้รองรับกับความต้องการของการนำมาใช้กับงานที่มีความซับซ้อนมากขึ้นเรื่อยๆ ของนั้น สิ่งสำคัญที่สุดย่อมขึ้นอยู่กับประสิทธิภาพของชิ้นส่วนแต่ละชิ้นที่อยู่ในอุปกรณ์อิเล็กทรอนิกส์ โดยจะเป็นส่วนประกอบที่สำคัญอย่างมากซึ่งก็คือ วงจรรวม (IC, Integrated Circuit) หรือ ไมโครชิป สำหรับตัวอย่างของการพัฒนาเทคโนโลยีไอซีที่พบเห็นได้ชัดเจน เช่น เทคโนโลยีไมโครโพรเซสเซอร์ (MPU), ไมโครคอนโทรลเลอร์ (MCU), และหน่วยความจำ (memory) ซึ่งในท้องตลาดปัจจุบันนี้มีไอซีหลายชนิดให้เลือกนำมาใช้งานกัน ตามความเหมาะสมทั้งด้านราคาและประสิทธิภาพ โดยเฉพาะอย่างยิ่งไอซีที่มีประสิทธิภาพสูงๆ ที่ใช้ในงานเฉพาะด้านหรือที่เรียกว่า ASIC (Application Specific Integrated Circuit) ซึ่งจะสามารถจัดแบ่งตามโครงสร้างออกได้เป็น 2 กลุ่มใหญ่ๆ คือ Field Programmable และ Mask Programmable ดังแสดงในรูป 6.1



รูปที่ 6.1 การแบ่งกลุ่มของวงจรรวม ASIC

ถ้าพิจารณาถึงข้อดี ในเชิงการออกแบบและการผลิต ไอซีตระกูล Mask programmable นั้น อาจจะทำให้ได้ชิป ASIC ที่มีประสิทธิภาพสูงกว่า แต่ต้องมีการส่งวงจรไปทำหน้ากา (Mask set) เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเอกสารยังต่างประเทศ จึงทำให้เสียบค่าใช้จ่ายสูงซึ่งเหมาะแก่การใช้งานที่ต้องการปริมาณสูงเพื่อเสียบค่าใช้จ่ายเริ่มต้นเท่านั้น สำหรับงานพัฒนาชิป ASIC ในปริมาณน้อยถึงปานกลางนั้น ควรใช้ชิป ASIC ชนิดโปรแกรมได้ ชนิดใดชนิดหนึ่งซึ่งสามารถหาซื้อได้ในราคาข่อมเขากว่า โดยผู้ใช้งานสามารถออกแบบและสร้างวงจรที่ต้องการใช้ลงในตัวอุปกรณ์ได้เอง โดยไม่ต้องไปผลิตที่โรงงาน จึงเป็นทางเลือกที่น่าสนใจ โดยเฉพาะอย่างยิ่ง เทคโนโลยีที่ก้าวหน้าไปมาก ทำให้ประสิทธิภาพของไอซีแบบโปรแกรมได้สามารถตอบสนองต่อความต้องการของงานที่มีความซับซ้อนสูงได้

6.1 เอฟพีจีเอ (FPGA: Field Programmable Gate Array)

FPGA เป็นอุปกรณ์ที่ถูกพัฒนาต่อจากอุปกรณ์แอสซีเอ ของบริษัทไซลิงค์ (Xilinx Inc.) โดยมีจุดเด่นที่ประสิทธิภาพการทำงานและปริมาณความหนาแน่นของจำนวนเกตสูง การใช้งานที่สะดวก สามารถกำหนดฟังก์ชันในการทำงานได้ตามความต้องการของผู้ใช้งาน โดยผ่าน software เพื่อจะทำการ โปรแกรมข้อมูลลงในชิป FPGA ให้ทำงานตามที่ออกแบบไว้โดยอัตโนมัติ ดังนั้นการออกแบบวงจร โดยใช้เอฟพีจีเอ สามารถออกแบบและทดสอบ ได้ภายในเวลาไม่นานเพียง 2-3 วันเท่านั้น ซึ่งตรงกันข้ามกับการออกแบบโดยใช้เกตอาเรย์ ซึ่งจะ ใช้เวลานานหลายสัปดาห์ การเปลี่ยนแปลงแก้ไขต้นแบบก็เช่นเดียวกัน จากประโยชน์ของเอฟพีจีเอ ดังที่กล่าวมานั้น ทำให้เกิดการประหยัดค่าใช้จ่ายเป็นอย่างมาก เพราะ ได้ลดความเสี่ยงในการที่จะต้องแก้ไขตัววงจรต้นแบบ การเลื่อนเวลาการออกผลิตภัณฑ์ และลดค่าเอ็นอาร์อี (NRE: Nonrecurring Engineering Cost) ลงไปด้วย

6.1.1 เปรียบเทียบกับไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ เป็นชิปประเภทใช้ประโยชน์ทั่วไปหมายความว่า สถาปัตยกรรม รวมถึงชุดคำสั่งทั้งหมดของชิปนั้นๆ ได้ถูกออกแบบและกำหนดมาเรียบร้อยแล้ว ผู้ใช้มีหน้าที่เพียงเรียบเรียงชุดคำสั่ง เพื่อให้ชิปปฏิบัติหน้าที่ตามที่ต้องการ ข้อจำกัดทางด้านสถาปัตยกรรมและชุดคำสั่งทำให้ไมโครคอนโทรลเลอร์ไม่ใช่คำตอบของงานหลายงานตัวอย่างเช่น ไมโครคอนโทรลเลอร์ที่ทำงานที่ความเร็ว 20 MHz ย่อมไม่สามารถใช้กับงานที่ต้องการความเร็วมากกว่านั้น ซึ่งอาจเป็นคุณสมบัติจำเป็นสำหรับงานในหลายๆด้าน เช่น โทรคมนาคมหรือ งานด้านประมวลผลภาพ (Image Processing) ทั้งหมดนี้ต่างกับการใช้ FPGA ที่ผู้ใช้เป็นคนออกแบบสถาปัตยกรรมทั้งหมดเอง โดยพื้นฐานเทคโนโลยี FPGA ในปัจจุบันสามารถทำงานที่สัญญาณนาฬิกามากกว่า 100 MHz เพราะฉะนั้นประสิทธิภาพจริงๆของชิป จะขึ้นกับสถาปัตยกรรมที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นผู้ใช้อาจสามารถออกแบบชุดคำสั่งใหม่ๆ ขึ้นเองเพื่อให้เหมาะกับการใช้งานนั้นๆ ได้ รวมถึงความสามารถในการรวบรวมวงจรดิจิทัลอื่นๆ นอกชิปเข้ามาอยู่รวมกันใน FPGA เดียวได้ FPGA จึงเป็นมากกว่าทางเลือกหนึ่งของไมโครคอนโทรลเลอร์แต่เป็นคำตอบที่ดีกว่าในงานออกแบบดิจิทัล

6.1.2 จุดเด่นของ FPGA

นอกจากระยะเวลาที่ใช้ในการออกแบบและพัฒนาจะเร็วใกล้เคียงกับการใช้ชิปประเภทประโยชน์ทั่วไป ขณะที่ประสิทธิภาพใกล้เคียงกับชิป ASIC แบบ full-custom ซึ่งหมายถึงชิปที่ถูกออกแบบและพัฒนาใหม่ทั้งหมดเฉพาะงานนั้น ซึ่งมีราคาแพงกว่ามากแล้ว ผู้ใช้ชิป FPGA ยังสามารถนำวงจรที่พร้อมใช้งานที่มีผู้ออกแบบไว้แล้ว ในรูปของทรัพย์สินทางปัญญาหรือ (IP-cores: Intellectual property cores) มาใช้ร่วมกับวงจรที่เราออกแบบเองได้ทันที ซึ่งเป็นการเพิ่มประสิทธิภาพและย่นเวลาในการพัฒนาชิป FPGA ของเราได้มาก

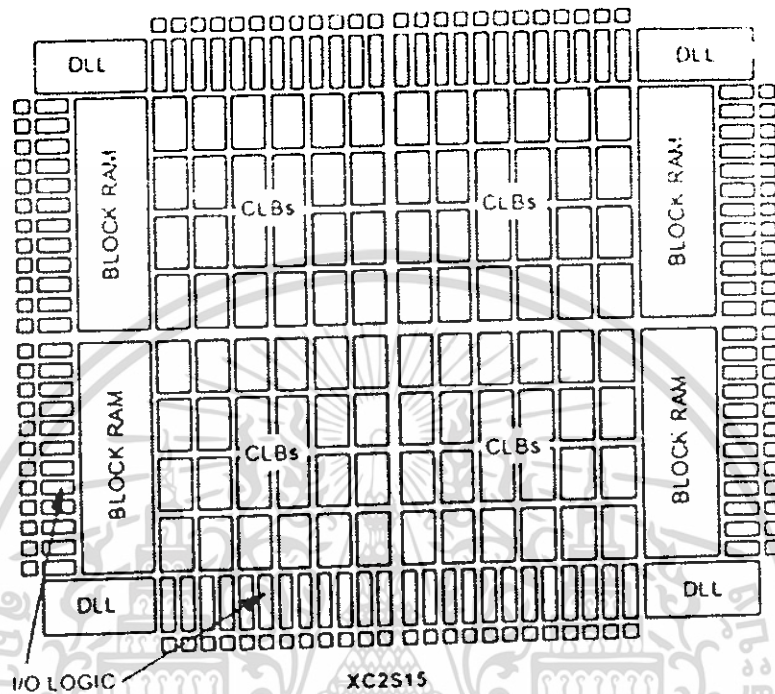
6.1.3 FPGA ในท้องตลาดและการเลือกใช้

ปัจจุบันมีผู้ผลิตชิป FPGA อยู่หลายบริษัทเช่น Xilinx Corp., Athela หรือ Actel โดยบริษัทที่มีขนาดใหญ่ที่สุดและมีส่วนแบ่งทางการตลาดสูงสุดคือ Xilinx กติ FPGA ของแต่ละบริษัทจะมีหลายตระกูลขึ้นอยู่กับราคาและขนาดความจุของชิป FPGA ซึ่งมักถูกวัดด้วยค่าจำนวนเกตเทียบเท่า โดยในปัจจุบัน FPGA มีขนาดตั้งแต่น้อยกว่า 10,000 เกต จนถึงมากกว่า 5 ล้านเกตขึ้นอยู่กับตระกูล เช่น Xilinx ตระกูล 9500 ซึ่งใช้เทคโนโลยี EEPROM มีขนาดความจุสูงสุดประมาณ 7,000 เกตขณะที่ Xilinx ตระกูล Virtex ซึ่งใช้เทคโนโลยี SRAM จะมีราคาแพงกว่าแต่มีขนาดความจุเพิ่มขึ้นกว่า 3 ล้านเกต เพื่อให้เห็นภาพเปรียบเทียบไมโครคอนโทรลเลอร์ตระกูล 8051 มีขนาดเกตประมาณ 15,000 เกต ขนาดความจุเกต ที่มากกว่าหมายถึงศักยภาพในการใช้งาน FPGA ตัวนั้นสำหรับงานที่มีความซับซ้อนมากกว่า

6.2 รายละเอียดสถาปัตยกรรมของ FPGA ตระกูล Spartan II

รูปที่ 6.2 แสดงบล็อกไดอะแกรมของโครงสร้างชิป FPGA ตระกูล Spartan II ประกอบด้วย Configurable Logic Block (CLBs) ซึ่งเป็นบล็อกลอจิกวางโครงแบบได้ ที่มีลักษณะสมมาตรอยู่ชิดหุ่นได้และ โปรแกรมได้ บล็อก CLB ทั้งหมดจะถูกล้อมรอบด้วย I/O Blocks (IOBs) โปรแกรมได้ บล็อกต่างๆเหล่านี้เชื่อมต่อกันด้วยแหล่งการเชื่อมโยง (Routing Resources) อันทรงประสิทธิภาพ

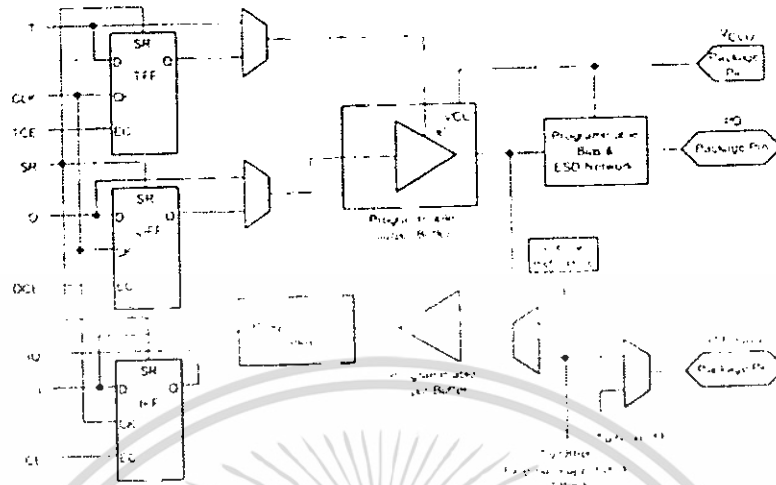
สถาปัตยกรรมนี้มีความสามารถรองรับฟังก์ชันขั้นสูง อย่างบล็อก RAM และบล็อกสัญญาณนาฬิกาควบคุม (Clock control block)



รูปที่ 6.2 บล็อกไออะแกรมของโครงสร้าง FPGA ตระกูล Spartan II

6.2.1 บล็อกอินพุต/เอาต์พุต (IOBs)

รูปที่ 6.3 แสดงบล็อกอินพุต เอาต์พุต (IOB) ของ โครงสร้าง FPGA ตระกูล Spartan II ซึ่งรองรับมาตรฐานการให้สัญญาณ I/O ความเร็วสูงสุดได้ถึง 16 มาตรฐาน เช่น LVCMOS, HTSL, SSTL และ GTL I/O ความเร็วสูงนี้สามารถรองรับหน่วยความจำ และการเชื่อมต่อของบัส (bus interface) ที่ทันสมัยได้หลายแบบรีจิสเตอร์ IOB ทั้งสามตัวทำหน้าที่เป็น edge-trigger D flip-flop (ฟลิปฟล็อปแบบประวิงชนิดที่ใช้ขอบสัญญาณนาฬิกาจุดชนวนการทำงาน) หรือ ไม่ก็เป็น Level-sensitive latch (แลตช์แบบไวต์ระดับสัญญาณนาฬิกา)

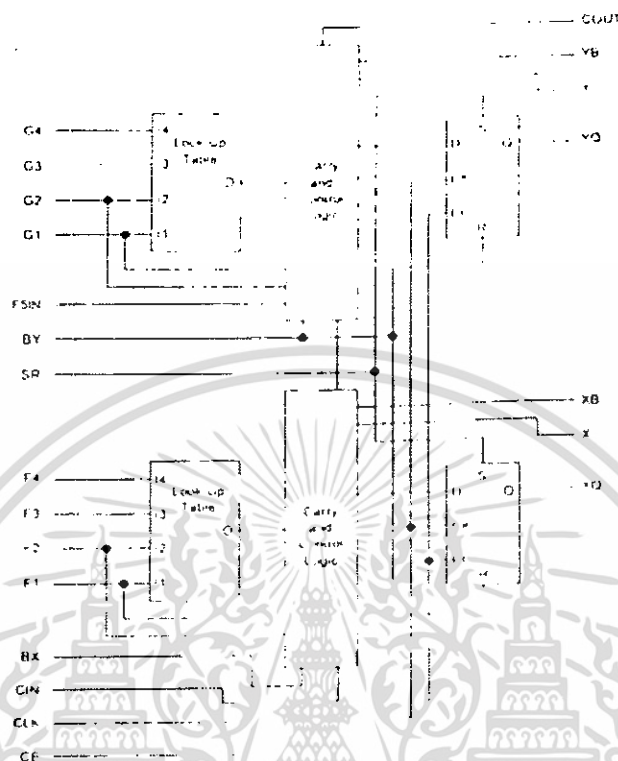


รูปที่ 6.3 บล็อกอินพุท/เอาต์พุทของ โครงสร้าง FPGA ตระกูล Spartan II

6.2.2 ลอจิกเซลล์ (Logic cells)

ลอจิกเซลล์เป็นส่วนประกอบพื้นฐานสำหรับ CLB ใน Spartan II ดังแสดงในรูปที่ 6.4 ในเซลล์ลอจิกหนึ่งมีตัวกำเนิดฟังก์ชันแบบ 4 input จำนวน 1 ตัว พร้อมลอจิกทอด (carry logic) และส่วนสำรองข้อมูล (storage element) เอาต์พุทจากตัวกำเนิดฟังก์ชันในแต่ละ LC เป็นสัญญาณเข้าทั้ง CLB output และ D input ของฟลิปฟล็อป ใน CLB แต่ละบล็อกของ Spartan II จะประกอบด้วย LC จำนวน 4 เซลล์ จัดเรียงใน slice นอกจาก LC พื้นฐานทั้งสี่แล้ว Spartan II CLB ยังมีลอจิกที่รวมเอาตัวกำเนิดฟังก์ชันที่ให้ฟังก์ชันที่ให้ฟังก์ชัน 5-6 input ไปด้วย ยังผลให้ CLB มีจำนวน LC ที่นับได้จริงเท่ากับ 4.5 เซลล์

ตัวกำเนิดฟังก์ชันของ Spartan II มีลักษณะเป็นตารางค้นหา (LUTs: Look-up Table) แบบ 4 input นอกจากเป็นตัวกำเนิดฟังก์ชันแล้ว LUT เหล่านี้สามารถทำหน้าที่เป็น 16 x 1-bits synchronous RAM หรือเป็น 16-bit shift register ที่เหมาะสำหรับบันทึกข้อมูลความเร็วสูงหรือข้อมูลที่ส่งออกเป็นชุดอย่างรวดเร็ว (burst-mode data) ซึ่งพบบ่อยในงานจำพวก Digital Signal Processing (DSP) ส่วนสำรองข้อมูลใน Spartan II slice อาจวางให้เป็นแบบ edge-triggered D flip-flops หรือเป็น level-sensitive latches ก็ได้



รูปที่ 6.4 บล็อก CLB slice ของ FPGA ตระกูล Spartan II (สมมาตรในแต่ละ CLB)

	Spartan-II	Spartan-XL	Spartan
ความหนาแน่น (เกต)	15K – 200K	5K – 40K	5K – 40K
สมรรถนะของ I/O	200 MHz	100 MHz	80 MHz
สถาปัตยกรรม	Virtex derivative	XC4000 derivative	XC4000 derivative
Block RAM	มี	ไม่มี	ไม่มี
Distributed RAM	มี	มี	มี
DLL	มี	ไม่มี	ไม่มี
มาตรฐาน I/O	16	4	4
Core Voltage	2.5 V	3.3 V	5 V
5 V Tolerance	มี	มี	มี
Process (micron)	0.18/0.22	0.25/0.35	0.35/0.5
Configuration mode	Serial, Parallel, JTAG	Serial, Express, JTAG	Serial, JTAG
Packages	VQ100, TQ/CS144, PQ208, FG256/456	PC84, VQ100, TQ/CS144, PQ208/240, BG256, CS280	PC84, VQ100, TQ144, PQ208/240, BG256

ตารางที่ 6.1 ตารางเปรียบเทียบคุณสมบัติ Spartan-II และ Spartan Series

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Feature	XC2S15	XC2S30	XC2S50	XC2s100	XC2S150	XC2S200
System Gates	15K	30K	50K	100K	150K	200K
Logic Cells	432	972	1728	2700	3888	5292
CLBs	96	216	384	600	864	1176
Block RAM (bits)	16K	24K	32K	40K	48K	56K
MAX I/O	86	132	176	196	260	284
Packages	VQ100 TQ144 CS144	VQ100 TQ144 CS144 PQ208	TQ144 PQ208 FG256	TQ144 PQ208 FG256 FG456	PQ208 FG256 FG456	PQ208 FG256 FG456

ตารางที่ 6.2 ตารางเปรียบเทียบคุณสมบัติตระกูล Spartan II series

อุปกรณ์	ขนาดความเร็ว		จำนวน pin / ชนิดแพคเกจ		ช่วงอุณหภูมิใช้งาน (Tj)	
	XC2S15	-5	มาตรฐาน	VQ100	100-pin Plastic Very Thin QFP	C = commercial
XC2S30	-6	สูง	CS144	144-ball Chip-Scale BGA	I = Industrial	-40°C ถึง 100°C
XC2S50			TQ144	144-pin Plastic Thin QFP		
XC2S100			PQ208	208-pin Plastic QFP		
XC2S150			FG256	256-ball Fine Pitch BGA		
XC2S200			FG456	456-ball Fine Pitch BGA		

ตารางที่ 6.3 ตารางแสดงรหัสหมายเลข FPGA ตระกูล Spartan II series

6.3 การออกแบบวงจรดิจิทัลในปัจจุบัน

รูปแบบการออกแบบวงจรดิจิทัลสามารถแบ่งออกได้เป็นหลายรูปแบบดังแสดงต่อไปนี้

6.3.1 การออกแบบโดยใช้การวาดผังวงจร (Schematic design)

ในอดีตการออกแบบวงจรดิจิทัลที่เราคุ้นกัน จะเป็นการออกแบบวงจรในระดับลอจิกเกต (Logic level) โดยอาศัยโปรแกรมช่วยในการวาดผังวงจรซึ่งการออกแบบวงจรในลักษณะนี้ค่อนข้างใช้เวลามากในการออกแบบ เนื่องจากการออกแบบทุกอย่างที่ผู้ออกแบบจะต้องออกแบบเองทั้งหมด ซึ่งถ้าวงจรที่ต้องการออกแบบมีความซับซ้อนสูง ก็จะทำให้การออกแบบทำได้ยาก ซึ่งเป็นข้อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำกัดให้ผู้ออกแบบไม่สามารถออกแบบวงจรดิจิทัลที่มีความซับซ้อนสูงได้ เช่น การออกแบบซีพียู และไมโครคอนโทรลเลอร์ เป็นต้น

6.3.2 การออกแบบโดยใช้ภาษาระดับสูง (HDL)

ในปัจจุบันเทคโนโลยีการออกแบบวงจรดิจิทัล ได้พัฒนาสูงขึ้นและได้มีการพัฒนาภาษาที่ใช้สำหรับออกแบบวงจรดิจิทัล พร้อมกระบวนการออกแบบแนวใหม่ ที่จะทำให้ผู้ออกแบบไม่ถูกจำกัดในการออกแบบวงจรอีกต่อไป ซึ่งการออกแบบวงจรดิจิทัลแนวใหม่นี้ จะใช้การเขียนภาษาบรรยายพฤติกรรมฮาร์ดแวร์ (HDL: Hardware Description Language) มาใช้ในการออกแบบวงจร โดยภาษาจะบรรยายพฤติกรรมฮาร์ดแวร์ที่เป็นมาตรฐานซึ่งมีอยู่ 2 ภาษาที่นิยมใช้ด้วยกันคือภาษา Verilog และ VHDL โดยถ้านำมาเปรียบเทียบกับในโลกของการออกแบบซอฟต์แวร์ก็เปรียบได้กับภาษา C และภาษา Pascal นั่นเอง โดยที่ภาษา Verilog จะมีโครงสร้างคล้ายกับภาษา C ส่วนภาษา VHDL จะมีโครงสร้างคล้ายกับภาษา Pascal

การออกแบบวงจรโดยใช้ภาษา HDL จะเป็นกระบวนการออกแบบที่เรียกว่า Top-Down Design โดยกระบวนการดังกล่าว จะอาศัยซอฟต์แวร์ในการช่วยออกแบบ ซึ่งเราเรียกว่าซอฟต์แวร์จำพวกนี้ว่า EDA: Electronics Design Automation ที่จะทำหน้าที่ช่วยในการจำลองการทำงานจากโค้ดที่เขียนขึ้นและช่วยในการนำโค้ดที่ผ่านการจำลองการทำงาน ไปสังเคราะห์ (Synthesis) ให้เป็นแผนผังวงจรระดับลอจิกเกต ในรูปแบบของเทคโนโลยีที่ผู้ออกแบบนำไปใช้งาน ซึ่งในปัจจุบัน มีอุปกรณ์หรือชิปที่สามารถรองรับการออกแบบวงจรแบบใช้ภาษา HDL ในการออกแบบ เช่น ชิป CDPD (Complex Programmable Logic Devices) หรือชิป FPGA (Field Programmable Gate Array) ที่มีความจุของเกตภายในสูง สามารถออกแบบวงจรที่มีความซับซ้อนสูงได้มากกว่า 1 ล้านเกต หรือจะนำไปออกแบบในระดับของแอสสิก (ASIC : Application Specific Integrated Circuits)

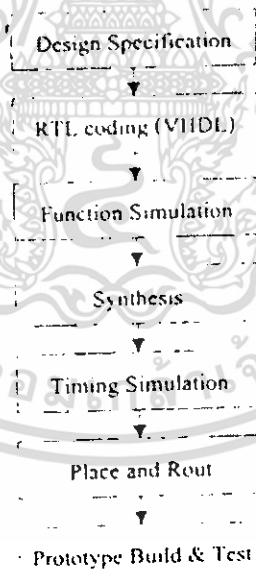
จะเห็นได้ว่าการออกแบบโดยใช้ภาษา HDL จะทำให้ผู้ออกแบบสามารถออกแบบวงจรที่ซับซ้อนได้อย่างสะดวกและรวดเร็วและที่สำคัญการออกแบบจะไม่ยึดติดกับเทคโนโลยีใดๆ สามารถนำกลับมาใช้ใหม่ หรือกล่าวได้ว่า ถ้าผู้ออกแบบต้องการเปลี่ยนเทคโนโลยีของวงจรที่ออกแบบ ก็เพียงสังเคราะห์วงจรใหม่ ก็จะได้วงจรเทคโนโลยีใหม่ทันที โดยไม่ต้องเสียเวลาในการวาดวงจรใหม่ เหมือนกับการออกแบบในลักษณะเดิมอีกต่อไป

6.4 การเขียนภาษา HDL

ดังกล่าวที่กล่าวไปแล้ว ในปัจจุบันแนวโน้มการออกแบบวงจรดิจิทัลขอบขนาดใหญ่ จะหันไปสู่การใช้ภาษาอธิบายฮาร์ดแวร์ ซึ่งมีบทบาทขึ้นตามลำดับ ด้วยเหตุผลหลายประการ อาทิเช่น

- ความสามารถในการอธิบายการทำงานของวงจร ที่มีขนาดใหญ่มาก
- ความสามารถในการอธิบายการในการนำกลับไปใช้งานได้อีกการของการออกแบบที่ถูกสร้างขึ้น
- ความสามารถในการอธิบายการนำกลับไปใช้งานได้กับหลายๆเทคโนโลยี

โดยโค้ดที่ถูกเขียนจะไม่ขึ้นอยู่กับเทคโนโลยีเหล่านั้น หรือเป็นเพียงส่วนน้อยเท่านั้นนอกจากนั้นยังมีซอฟต์แวร์ที่มีประสิทธิภาพอีกมากมายสำหรับใช้ในการสังเคราะห์ ในปัจจุบันภาษาที่ใช้อธิบายฮาร์ดแวร์จะเป็นที่นิยมใช้ และเป็นมาตรฐานในทางอุตสาหกรรมด้วยคือ ภาษา VHDL และ Verilog HDL โดยที่ภาษาทั้งสองก็มีจุดเด่นจุดด้อย และประสิทธิภาพในการใช้งานรวมถึงซอฟต์แวร์ที่ใช้สนับสนุนที่ไม่แตกต่างกัน โดยจะขอแนะนำวิธีการเขียนภาษา VHDL และ Verilog แต่เพียงคร่าวๆ เพื่อเป็นพื้นฐานในการออกแบบ



รูปที่ 6.5 ขั้นตอนการออกแบบวงจรด้วยภาษา HDL

6.4.1 ลักษณะเฉพาะในการออกแบบ (Design Specification)

รูปที่ 6.5 เป็นการแสดงการออกแบบวงจร โดยใช้ภาษา HDL จะเริ่มต้นด้วยการเขียนโค้ด หลังจากที่ได้มีการออกแบบสถาปัตยกรรมของวงจรอย่างคิดแล้ว เช่น อาจจะเริ่มมีการแบ่งวงจรรวม

ออกเป็นวงจรรย่อย เพื่อง่ายต่อการออกแบบตามหลักการ Top-Down Design Methodology ถ้าวงจร

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการเรียนเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือบล็อกย่อยไม่ซับซ้อนเกินไปและง่ายต่อการสังเคราะห์วงจรในภายหลัง ก็สามารถเขียนอธิบายพฤติกรรมการทำงานของวงจรเหล่านั้น (Behavioral Description) เป็นโมเดลในภาษา HDL และเมื่อวงจรย่อยเหล่านี้เสร็จแล้ว จึงนำมาประกอบเข้าด้วยกัน โดยการเรียกใช้ส่วนประกอบ แต่อย่างไรก็ตาม ยังมีซอฟต์แวร์หลายตัวที่สนับสนุนการเขียนโค้ดภาษา HDL ซึ่งทำให้ง่ายขึ้นเวลาเขียนโค้ดในภาษา HDL รวมทั้งการคอมไพล์โค้ดเพื่อตรวจเช็คความถูกต้องตามหลักไวยากรณ์ด้วย ตัวอย่างของคอมไพเลอร์ภาษา HDL สำหรับเครื่องพีซีก็เช่น ซอฟต์แวร์ชื่อ ModelSim ของ Model Technology, Inc. เป็นต้น

6.4.2 การจำลองการทำงานฟังก์ชัน (Function simulation)

ขั้นตอนอีกขั้นตอนหนึ่งที่สำคัญในการออกแบบวงจร คือ การตรวจสอบว่าแบบจำลอง (Model) ในภาษา HDL ที่เราได้สร้างขึ้นนั้นสามารถทำหน้าที่ได้ตามที่กำหนดไว้หรือไม่ ซึ่งสามารถทำได้โดย การทำการจำลองการทำงาน เพื่อดูพฤติกรรมการทำงานของต้นแบบเหล่านั้น (Behavioral Simulation) และด้วยจุดประสงค์นี้เอง จะต้องมีการเขียนต้นแบบพิเศษสำหรับการทำการจำลองการทำงานขึ้นอีก ที่เรียกว่า HDL Test bench ซึ่งภายในจะมีการเรียกใช้หน่วยย่อยหรือส่วนประกอบที่ต้องการจะตรวจสอบพฤติกรรมการทำงาน และมีการกำหนดรูปแบบของสัญญาณขาเข้า (Input Signal) อันเป็นการกระตุ้น (Stimulus) และเมื่อทำการจำลองการทำงานสัญญาณขาออก (Output Signal) ของหน่วยย่อยนั้นก็จะเป็นปรากฏให้เห็นบนจอภาพ และสามารถนำไปเปรียบเทียบกับสัญญาณขาเข้าเพื่อตรวจสอบดูว่า การทำงานของหน่วยย่อย หรือวงจรที่ออกแบบแล้วถูกตรวจสอบ (Design Under Test หรือ DUT) เป็นไปตามที่คาดหวังไว้หรือไม่

6.4.3 การสังเคราะห์วงจร (Synthesis)

เมื่อได้ทำการตรวจสอบเป็นที่แน่ใจแล้วว่า ต้นแบบต่างๆที่สร้างขึ้นสามารถทำงานได้อย่างถูกต้อง ขั้นตอนต่อไปก็คือ การผ่าน โค้ดของต้นแบบเหล่านั้น ไปทำการสังเคราะห์วงจร เพื่อให้ได้เนตลิสต์ในระดับเกตออกมา เครื่องมือที่ใช้ในการสังเคราะห์โค้ดภาษา HDL (HDL Synthesis Tool) ตัวอย่างเช่น ซอฟต์แวร์ของบริษัทเมนเตอร์กราฟฟิกส์ (Mentor Graphics) ที่มีชื่อว่า “ลีโอนาร์โด สเปกตรัม” (Leonardo Spectrum) ซึ่งสามารถใช้ในการสังเคราะห์วงจรให้เป็นเอตลิก หรือ เอฟพีจีเอก็ได้เพราะมีไลบรารี ให้เลือกใช้ได้จากหลายบริษัทผู้ผลิต เนื่องจากเราจะสังเคราะห์วงจรสำหรับใช้งานกับเอฟพีจีเอของ Xilinx ดังนั้น ไลบรารีที่ใช้ก็จะเป็นของ Xilinx เช่น ไลบรารีของตระกูล Spartan II เป็นต้น

6.4.4 Timing simulation

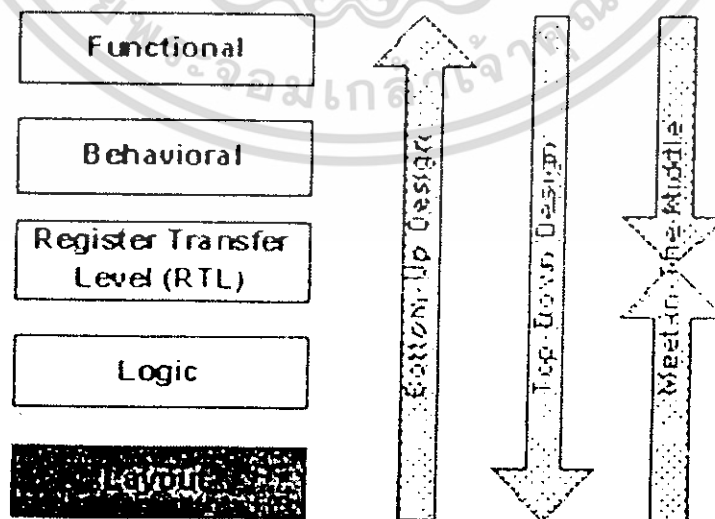
การทำการจำลองการทำงานในระดับพฤติกรรม หรือหลังจากการสังเคราะห์วงจร แต่ยังมีได้ผ่านขั้นตอนการวางและเชื่อมเส้นทางนั้น จะใช้ในการตรวจสอบความถูกต้องทางลอจิกของวงจรเท่านั้น แต่ยังไม่มียุทธศาสตร์ที่แน่นอนเกี่ยวกับเรื่องของเวลา (Timing) มากนัก เช่น ความล่าช้าของสัญญาณไฟฟ้าตามเส้นทางต่างๆภายในวงจร ระยะคาบของสัญญาณเวลาที่น้อยที่สุดเท่าที่จะสามารถใช้กับวงจรได้ เป็นต้น เพราะจะต้องให้ผ่านขั้นตอนการวางและเชื่อมเส้นทางก่อน ดังนั้นการทำการจำลองการทำงานเพื่อตรวจสอบการทำงานของวงจร ในอุปกรณ์เอพฟิซีเอโดยคำนึงถึงเรื่องไทม์มิง จะกระทำเป็นขั้นตอนสุดท้าย

6.4.5 Place & Route

ผลที่ได้จากการสังเคราะห์นั้นจะเป็นเน็ตลิสต์ในระดับเกต โดยสามารถนำไปผ่านขั้นตอนการแปลงให้เป็นลอจิกบล็อก และรวมถึงการวางและเชื่อมเส้นทาง (Place & Route) ภายในอุปกรณ์เอพฟิซีเอ โดยใช้ซอฟต์แวร์ Xilinx ISE ของ Xilinx ก็ได้

6.5 การออกแบบจากบนลงล่างและการออกแบบจากล่างขึ้นบน

ในการออกแบบด้วยเทคโนโลยี ASIC หรือ FPGA มีวิธีการและกระบวนการออกแบบอยู่ 2 วิธีคือ การออกแบบจากบนลงล่าง (Top-Down Design) และการออกแบบจากล่างขึ้นบน (Bottom-Up Design) ดังรูปที่ ซึ่งทั้งสองกระบวนการมีข้อดีและข้อเสียแตกต่างกัน



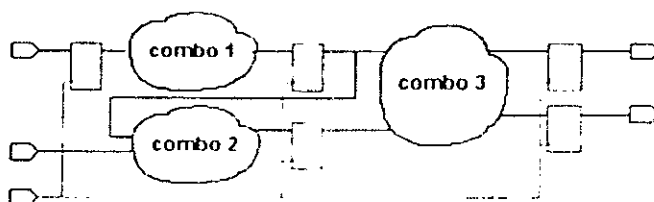
รูปที่ 6.6 กระบวนการออกแบบ (Design Methodology)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบจากบนลงล่าง (Top-Down Design) จะมองการออกแบบจากระดับบนสุดของวงจร หรือระบบที่ต้องการออกแบบ นั่นคือ ระดับของฟังก์ชัน (Functional level) การทำงานของวงจรที่ต้องการออกแบบ และลงลึกในรายละเอียดระดับล่างของแต่ละฟังก์ชัน ในระดับของพฤติกรรม (Behavioral level) การทำงานของแต่ละฟังก์ชัน และการออกแบบรายละเอียดของแต่ละฟังก์ชันในระดับรีจิสเตอร์ (RTL) ของแต่ละฟังก์ชัน และค่อยวิเคราะห์ลึกลงไป ในรายละเอียดระดับลอจิก (Logic Level) และระดับสุดท้าย ซึ่งเป็นระดับล่างที่สุดของการออกแบบ คือระดับผังภูมิวงจร (Layout) ซึ่งผู้ออกแบบจะไม่ทราบรายละเอียดของวงจรที่ออกแบบ จนกว่าจะถึงระดับสุดท้ายของการออกแบบ ซึ่งจะแตกต่างจากการออกแบบจากล่างขึ้นบน (Bottom-Up Design) ที่จะเริ่มออกแบบจากระดับล่างสุด และนำแต่ละส่วนประกอบกัน ขึ้นเป็นบล็อกในระดับบน เพื่อให้วงจรสามารถทำหน้าที่ตามฟังก์ชันนั้นๆ ได้ กระบวนการออกแบบลักษณะที่ผู้ออกแบบจะทราบรายละเอียดของวงจรทั้งหมด แต่ข้อเสียของกระบวนการนี้ คือผู้ออกแบบจะต้องทำการเลือกเทคโนโลยีใดๆ ก่อนนำไปออกแบบวงจร ซึ่งถ้ามีการเปลี่ยนเทคโนโลยี จะต้องทำการออกแบบวงจรใหม่ แต่ในทางกลับกันผู้ออกแบบที่ใช้กระบวนการออกแบบจากบนลงล่าง จะไม่ยึดติดกับเทคโนโลยี ในการออกแบบวงจร เพราะการออกแบบลักษณะนี้วงจรจะอยู่ในรูปแบบของโมเดลภาษา HDL ซึ่งถ้าต้องการใช้เทคโนโลยีอะไร ก็เพียงนำไลบรารีของเทคโนโลยีมาสังเคราะห์ เป็นโมเดลของวงจรในระดับล่างต่อไปเท่านั้น นี่คือการแตกต่างระหว่างกระบวนการออกแบบ จากบนลงล่างกับกระบวนการออกแบบจากล่างขึ้นบน

6.6 RTL (Register Transfer Level)

RTL คือการเขียนบรรยายพฤติกรรมของวงจรในระดับของรีจิสเตอร์และวงจรคอมบิเนชัน การออกแบบในระดับนี้ นักออกแบบต้องนึกถึงรูปแบบการประมวลผล ไลน์มิ่งและการเชื่อมต่อของสัญญาณระหว่าง กลุ่มวงจรคอมบิเนชันและรีจิสเตอร์ โดยที่ไม่จำเป็นต้องพิจารณาพฤติกรรมวงจรระดับลอจิก หรือระดับทรานซิสเตอร์ รูปที่ แสดงบล็อกไดอะแกรมของ โครงสร้างของ RTL



รูปที่ 6.7 ตัวอย่าง Register Transfer Level บล็อกไดอะแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการออกแบบวงจรดิจิทัลด้วย HDL ส่วนใหญ่ จะทำการเขียนบรรยายพฤติกรรมวงจรในระดับ RTL เนื่องจากว่ามีความสะดวกในการออกแบบ และให้วงจรที่มีประสิทธิภาพมากกว่า การเขียนบรรยายในระดับลอจิก ซึ่งใช้เวลาออกแบบนาน และหาข้อผิดพลาดลำบาก หรือในระดับพฤติกรรม ออกแบบง่ายแต่วงจรไม่ให้ผลดีที่สุด (Optimize) ดังนั้นการออกแบบในระดับ RTL จะมีความสะดวกในขั้นตอนของการสังเคราะห์วงจร การวิเคราะห์เวลา (Timing Analysis) ซึ่งเป็นการวิเคราะห์แก้ปัญหาของค่าเวลาดำเนินวงจร รวมทั้งการแบ่งวงจรแยกออกจากกันเป็นบล็อกย่อย (Partitioning) สำหรับการสังเคราะห์ (Synthesis) เพื่อให้ผลลัพธ์ที่ดีที่สุด และยังสามารถทำความเข้าใจได้ไม่ยากนักสำหรับนักออกแบบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

การออกแบบชิ้นงาน

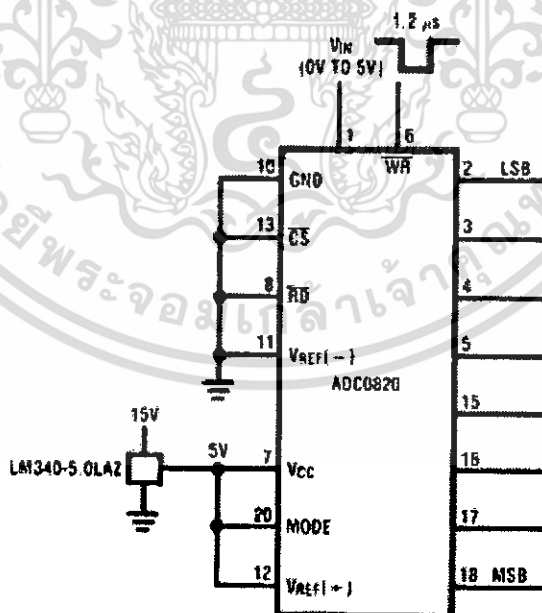
ในโครงการนี้จะทำการสร้างเฉพาะตัวกรองความถี่ซึ่งเป็นสิ่งที่สำคัญที่สุดของอีควอไลเซอร์ ประกอบด้วยส่วนต่างๆดังนี้ วงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล ตัวกรองสัญญาณ และวงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก



รูปที่ 7.1 บล็อกโคอะแกรมของอีควอไลเซอร์

7.1. วงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล

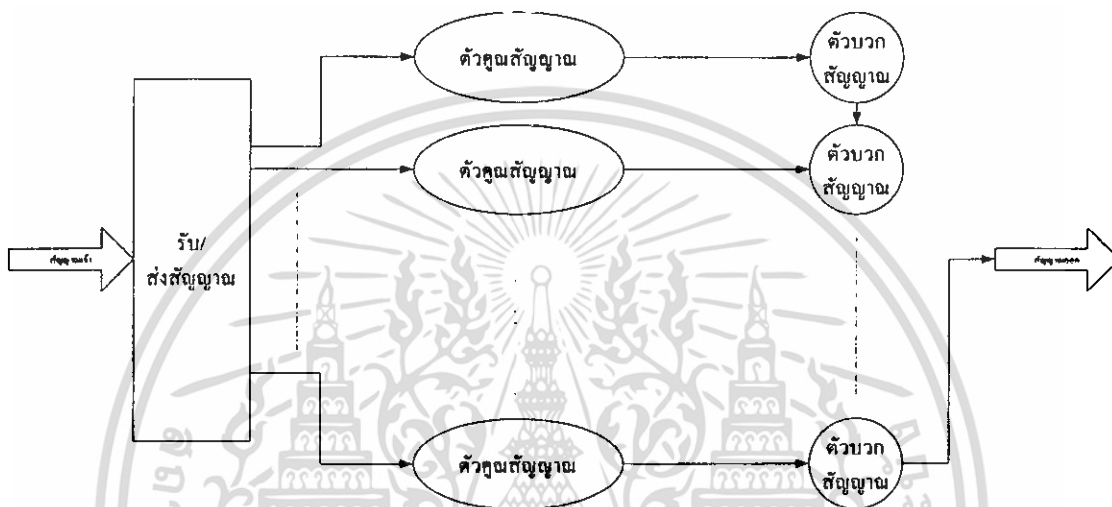
ในส่วนของวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณเชิงเลข โดยใช้ไอซีเบอร์ ADC0820 ต่อวงจรดังรูป 7.2 ได้เอาที่พุดออกมาเป็นเลขฐานสอง 8 บิตแบบวันคอมพลิเมนต์



รูปที่ 7.2 รูปวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณเชิงเลข

7.2 ตัวกรองสัญญาณ

ถือได้ว่าเป็นส่วนสำคัญที่สุดสำหรับ อีควอไลเซอร์เนื่องจากคุณภาพของตัวกรองสัญญาณ (ขึ้นอยู่กับปริมาณลำดับของตัวกรองสัญญาณ) เป็นตัวกำหนดคุณภาพของอีควอไลเซอร์ โดยในการสร้างจะใช้ภาษา VHDL โปรแกรมในโปรแกรม Webpack Xilinx 7.1



รูปที่ 7.3 บล็อกไออะแกรมของตัวกรองสัญญาณ

ตัวกรองสัญญาณเชิงเลข จะทำการคูณสัญญาณแล้วเลื่อนสัญญาณไปเรื่อยๆ พร้อมกับรวมสัญญาณทั้งหมดเข้าด้วยกันหรือที่เรียกว่า การคอนโวลูชัน (Convolution) ตามรูปที่ 7.3

ส่วนประกอบของตัวกรองสัญญาณ

7.2.1 แปลงเลขฐานสองแบบ วันคอมพลิเมนต์เป็นทวูคอมพลิเมนต์

สัญญาณที่ได้จากวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล จะได้เป็นเลขฐานสองแบบวันคอมพลิเมนต์ จะต้องทำการแปลงเป็นเลขฐานสองแบบทวูคอมพลิเมนต์ เพื่อให้สามารถทำการคูณกับเลขสัมประสิทธิ์ของตัวกรองสัญญาณได้ เนื่องจากว่าค่าสัมประสิทธิ์มีทั้งค่าที่เป็นทศนิยมและมีค่าติดลบด้วย จึงต้องใช้เลขฐานสองแบบทวูคอมพลิเมนต์ในการแสดงค่าดังกล่าว

```

Entity one_stotwo_s is
Port ( in_1s : in std_logic_vector(7 downto 0);
      out_2s : out std_logic_vector(7 downto 0));
End one_stotwo_s;
Architecture Behavioral of one_stotwo_s is
Begin

```

```

    Process (in_1s)
    begin
        out_2s(7) <= '0';
        out_2s(6) <= in_1s(7);
        out_2s(5) <= in_1s(6);
        out_2s(4) <= in_1s(5);
        out_2s(3) <= in_1s(4);
        out_2s(2) <= in_1s(3);
        out_2s(1) <= in_1s(2);
        out_2s(0) <= in_1s(1);
    end process;

```

```

End Behavioral;

```

7.2.2 ตัวลำดับการจ่ายสัญญาณ (Splitter)

การป้อนสัญญาณอินพุตเพื่อนำไปคูณค่าสัมประสิทธิ์ จะเป็นการนำอินพุตไปคูณกับสัมประสิทธิ์ตัวแรก แล้วจึงเลื่อนอินพุตนั้น ไปคูณกับสัมประสิทธิ์ตัวที่สอง ในขณะที่เดียวกันก็รับอินพุตใหม่มาคูณกับสัมประสิทธิ์ตัวแรก ผลลัพธ์จะ ได้จากการรวมผลคูณเข้าด้วยกัน แล้วจึงเลื่อนอินพุตไปอย่างนี้เรื่อยๆ ในที่นี้จะยกตัวอย่างเพียง 5 สัมประสิทธิ์เท่านั้น

```

Entity splitter is
Port ( in_2s : in std_logic_vector(7 downto 0);
      clk   : in std_logic;
      outs1 : out std_logic_vector(7 downto 0);
      outs2 : out std_logic_vector(7 downto 0);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
outs3 : out std_logic_vector(7 downto 0);
```

```
outs4 : out std_logic_vector(7 downto 0);
```

```
outs5 : out std_logic_vector(7 downto 0);
```

```
End splitter;
```

Architecture Behavioral of splitter is

```
Signal out1,out2,out3,out4,out5 : std_logic_vector(7 downto 0) := "00000000";
```

```
Begin
```

```
Process (in_2s,clk)
```

```
begin
```

```
if clk = '1' then
```

```
out5 <= out4;
```

```
out4 <= out3;
```

```
out3 <= out2;
```

```
out2 <= out1;
```

```
out1 <= in_2s;
```

```
outs1 <= out1;
```

```
outs2 <= out2;
```

```
outs3 <= out3;
```

```
outs4 <= out4;
```

```
outs5 <= out5;
```

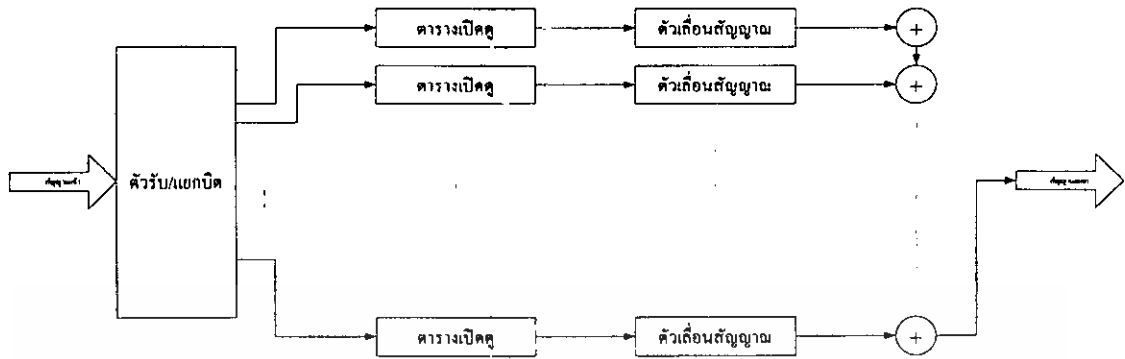
```
end if;
```

```
end process;
```

```
End Behavioral;
```

7.2.3 ตัวคูณ (Multiplier)

เป็นส่วนที่ทำการคูณสัญญาณกับค่าสัมประสิทธิ์ของตัวกรองสัญญาณที่ได้มาจากโปรแกรม Matlab และแปลงเป็นเลขฐานสองแบบทศนิยมพลีเมนต์ด้วยมือตามตาราง



รูปที่ 7.4 บล็อกไดอะแกรมของตัวคูณสัญญาณ

การคูณสัญญาณนี้ เราจะใช้การคูณสัญญาณแบบใช้ ตารางเปิดดู (Look up table) สัญญาณที่เข้ามาจะถูกแยกออกเป็นแต่ละบิต แล้วจึงถูกนำไปเทียบกับตารางเปิดดู หลังจากนั้นสัญญาณที่มาจากตารางเปิดดูจะถูกเลื่อนไป สุดท้ายสัญญาณทั้งหมดจะถูกนำมารวมกันแล้วจึงได้เป็นผลคูณ ดังรูปที่ 7.4 ในส่วนของตัวคูณสัญญาณจะประกอบไปด้วย

7.2.3.1 ตัวแยกบิต เป็นตัวแยกสัญญาณให้เป็นหนึ่งบิต แล้วจึงนำไปเข้าตารางเปิดดู

Entity splitmulti_1 is

Port(input_split : in std_logic_vector(7 downto 0);

load : in std_logic;

output7_split ,output6_split ,output5_split : out std_logic;

output4_split ,output3_split ,output2_split : out std_logic;

output1_split ,output0_split : out std_logic);

End splitmulti_1 ;

Architecture Behavioral of splitmulti_1 is

Signal a_bus : std_logic_vector(7 downto 0);

Begin

Process (input_split, load)

Begin

if (load'event and load='1') then

a_bus <= input_split;

output7_split <= a_bus(7);

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

output6_split <= a_bus(6);
output5_split <= a_bus(5);
output4_split <= a_bus(4);
output3_split <= a_bus(3);
output2_split <= a_bus(2);
output1_split <= a_bus(1);
output0_split <= a_bus(0);
end if;
end process;

```

End Behavioral;

7.2.3.2 ตารางเปิดดู (Look up Table) เป็นส่วนที่นำอินพุตหนึ่งบิตเข้ามา ถ้าเป็นอินพุตมีค่าเป็น 1 จะได้อาท์พุท เป็นสามประสิทธิ์ แต่ถ้าอินพุตมีค่าเป็น 0 จะได้อาท์พุทเป็น "00000000"

Package lut_pack_210 is

```

Subtype lut_word is std_logic_vector(7 downto 0);
Type lut_table is array (0 to 1) of lut_word;
Constant lut_data_210 : lut_table := ( "00000000",
                                         "11100001" );

```

End;

และ

Entity lut_21 is

```

Port (addr_in : in std_logic;
      data    : out lut_word);

```

End lut_21;

Architecture behav of lut_21 is

Begin

Process(addr_in)

Variable addr : std_logic;

Begin

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    addr := addr_in;
    data <= lut_data_21(conv_integer(addr));
End Process;
End behav;

```

7.2.3.3 ตัวเลื่อนสัญญาณ (Shift) จะเป็นการเลื่อนข้อมูลแต่ละหลักก่อนนำมารวมกัน

```

Entity shift_4 is
Port (input_shift      : in std_logic_vector(7 downto 0);
      output_shift     : out std_logic_vector(14 downto 0):="0000000000000000");
End shift_4;
Architecture Behavioral of shift_4 is
Begin
    Process(input_shift)
    Variable bf_out : std_logic_vector(3 downto 0);
    Variable s      : std_logic_vector(14 downto 0);
    Begin
        bf_out(0) := input_shift(7);
        bf_out(1) := input_shift(7);
        bf_out(2) := input_shift(7);
        bf_out(3) := input_shift(7);
        s := bf_out & input_shift & "000";
        output_shift <= s;
    end process;
End Behavioral;

```

7.2.3.4 ตัวรวมสัญญาณ (Sum) เป็นส่วนรวมค่าที่ได้จากการคูณบิตกับสัมประสิทธิ์แต่ละตัวมาบวกกันรวมเป็นสัญญาณที่ผ่านการคูณแล้ว

Entity add_1 is

```
Port( in_1 : in std_logic_vector(7 downto 0);
      in_2 : in std_logic_vector(14 downto 0);
      addout : out std_logic_vector(14 downto 0));
End;
```

Architecture behav of add_1 is

Begin

Process (in_1,in_2)

Variable s_sub : std_logic_vector(7 downto 0);

Variable s_xxx,sum : std_logic_vector(14 downto 0);

begin

s_sub:=(in_1 xor "11111111");

s_sub := s_sub+1;

s_xxx := s_sub & "0000000";

sum(14 downto 0):= in_2(14 downto 0)+ s_xxx(14 downto 0);

addout<=sum;

end Process ;

End behav;

และ

Entity add_2 is

```
Port (in_1,in_2 : in std_logic_vector(14 downto 0);
```

```
      addout : out std_logic_vector(14 downto 0));
```

End add_2;

Architecture behav of add_2 is

Begin

Process(in_1,in_2)

begin

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    addout <= in_1(14 downto 0) + in_2(14 downto 0);
  End Process;
End behav;
และ
Entity addfin is
Port (input_shift      : in std_logic_vector(14 downto 0);
      output_shift     : out std_logic_vector(7 downto 0));
End addfin;
Architecture Behavioral of addfin is
Begin
  Process(input_shift)
  Variable bf_out : std_logic_vector(7 downto 0);
  begin
    bf_out(0) := input_shift(7);
    bf_out(1) := input_shift(8);
    bf_out(2) := input_shift(9);
    bf_out(3) := input_shift(10);
    bf_out(4) := input_shift(11);
    bf_out(5) := input_shift(12);
    bf_out(6) := input_shift(13);
    bf_out(7) := input_shift(14);
    output_shift <= bf_out;
  end process;
End Behavioral;

```

7.2.4 แปลงเลขฐานสองแบบ ทูคอมพลิเมนต์เป็นวันคอมพลิเมนต์

การแปลงสัญณาคติจิตอลเป็นสัญณาคอนาถ็อก จะต้องใช้เลขฐานสองแบบทูคอมพลิเมนต์ จึงต้องทำการแปลงกลับจากเลขฐานสองแบบวันคอมพลิเมนต์เป็นเลขฐานสองแบบทูคอมพลิเมนต์

```

Entity two_stoone_s is
Port ( in_2s : in std_logic_vector(7 downto 0);
      out_1s : out std_logic_vector(7 downto 0));
End two_stoone_s;

Architecture Behavioral of two_stoone_s is
Begin
Process (in_2s)
begin
out_1s(7) <= in_2s(7) xor '1';
out_1s(6) <= in_2s(6);
out_1s(5) <= in_2s(5);
out_1s(4) <= in_2s(4);
out_1s(3) <= in_2s(3);
out_1s(2) <= in_2s(2);
out_1s(1) <= in_2s(1);
out_1s(0) <= in_2s(0);
end process;
End Behavioral;

```

7.2.5 ตัวรวมสัญญาณ (Sum) เป็นส่วนรวมค่าที่ได้จากการคูณสัญญาณกับสัมประสิทธิ์แต่ละคัมมาบวกกันรวมเป็นสัญญาณที่ผ่านการตัวกรองแล้ว ซึ่งจะยกตัวอย่างเพียงการรวมสัญญาณ 5 อันเท่านั้น

```

Entity sum is
Port ( in1 : in std_logic_vector(7 downto 0);
      in2 : in std_logic_vector(7 downto 0);
      in3 : in std_logic_vector(7 downto 0);
      in4 : in std_logic_vector(7 downto 0);
      in5 : in std_logic_vector(7 downto 0);
      clk : in std_logic;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

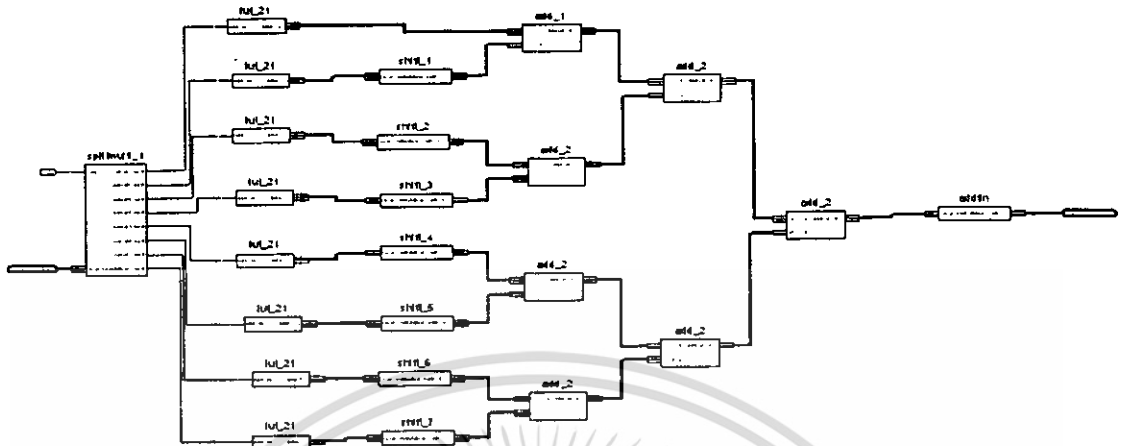
    out1 : out std_logic_vector(7 downto 0));

End sum;

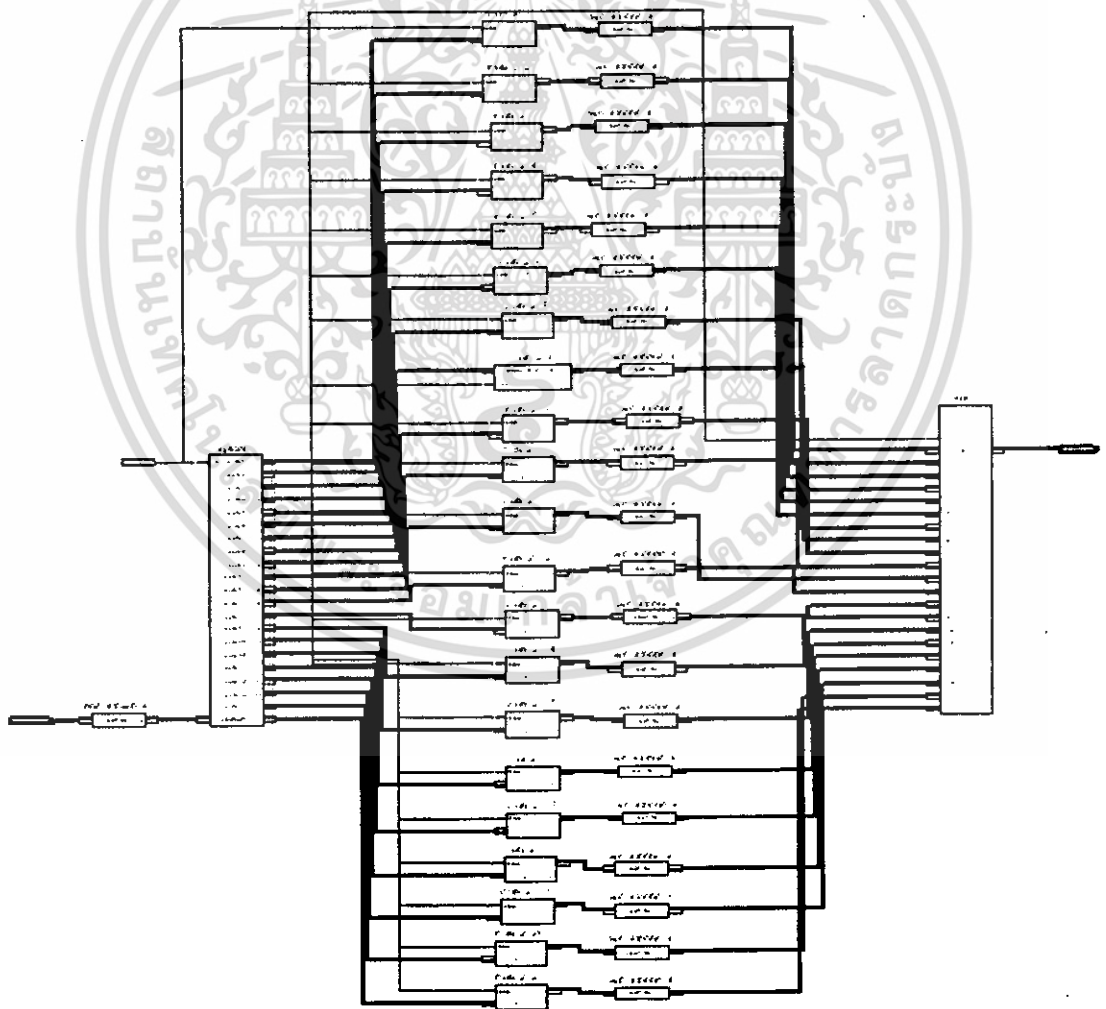
Architecture Behavioral of sum is
Signal temp1,temp2,temp3,temp4,temp5 : std_logic_vector(10 downto 0);
Signal a : std_logic_vector(10 downto 0);
Begin
    Process (clk)
    Begin
        if clk = '1' then
            temp1 <= "000" & in1;
            temp2 <= "000" & in2;
            temp3 <= "000" & in3;
            temp4 <= "000" & in4;
            temp5 <= "000" & in5;
            a <= temp1 + temp2 + temp3 + temp4 + temp5;
            if a(10) = '1' or a(9) = '1' or a(8) = '1' then
                out1 <= "11111111";
            else
                out1(0) <= a(0);
                out1(1) <= a(1);
                out1(2) <= a(2);
                out1(3) <= a(3);
                out1(4) <= a(4);
                out1(5) <= a(5);
                out1(6) <= a(6);
                out1(7) <= a(7);
            end if;
        end if;
    end process;
End Behavioral;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.5 โปรแกรมของตัวคูณสัญญาณ



รูปที่ 7.6 โปรแกรมของตัวกรองสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าพารามิเตอร์ของตัวกรองสัญญาณที่ทำการสร้าง

Digital Filter FIR Window Blackman

Order 21

Sampling Frequency 48 KHz

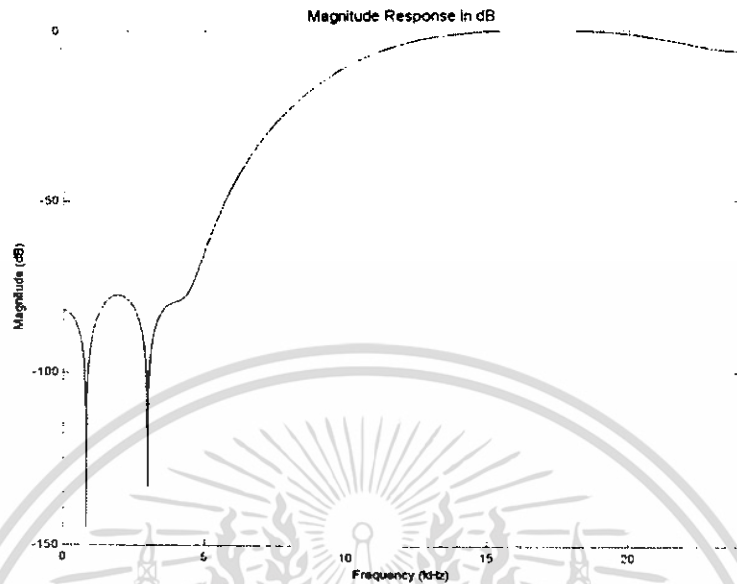
Cut-off Frequency 1 = 11 KHz

Cut-off Frequency 2 = 22.4 KHz

ลำดับ	ค่าสัมประสิทธิ์	เลขฐานสองแบบทอคอมพลิเมนต์
1	0.0000000000000000	0000 0000
2	0.000184800512604	0000 0000
3	-0.000205596822213	1111 1111
4	0.007391633297504	0000 0001
5	-0.017661407206292	1111 1110
6	0.001572992065630	0000 0000
7	-0.009863795077781	1111 1110
8	0.110537981769821	0000 1110
9	-0.089955927461926	1111 0100
10	-0.239492356407215	1110 0001
11	0.4750000000000000	0101 1101

ตารางที่ 7.1 ค่าสัมประสิทธิ์ของตัวกรองสัญญาณ

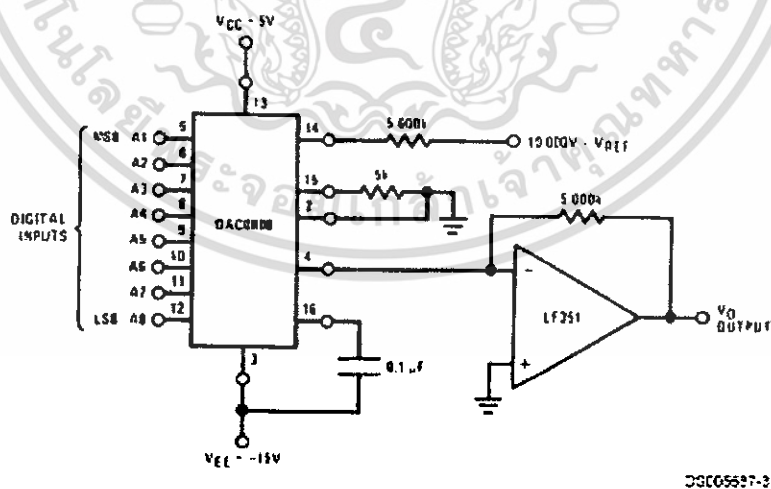
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.7 ค่าผลการตอบสนองทางความถี่ที่ได้จากการออกแบบตัวกรองสัญญาณด้วยโปรแกรม Matlab

7.3 วงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก

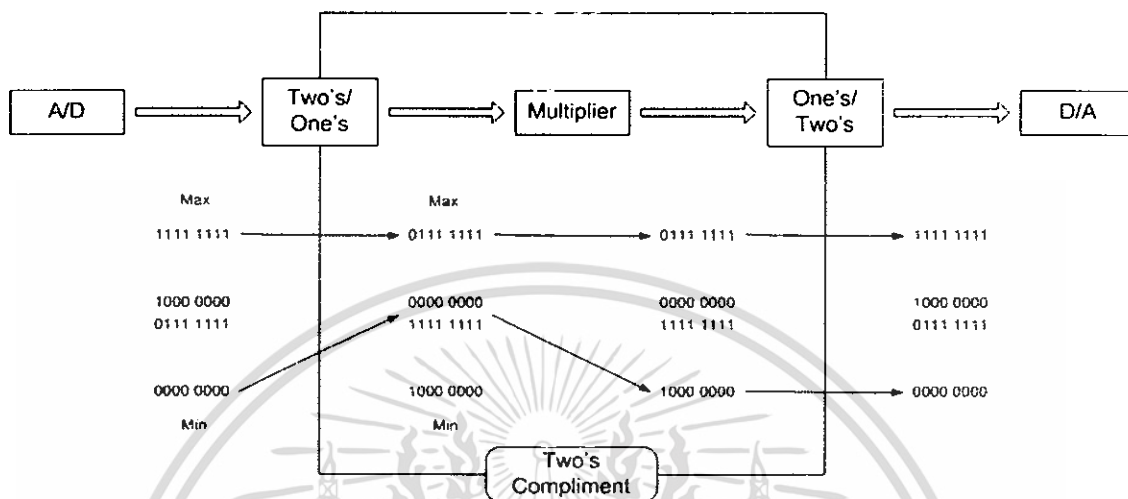
ส่วนวงจรแปลงสัญญาณเชิงเลขเป็นสัญญาณอนาล็อกจะใช้วิธีอาร์-ทูอาร์ (R-2R Ladder) ใช้ไอซีเบอร์ DAC0808 ค่่วงจรคังรูป



รูปที่ 7.8 รูปวงจรแปลงสัญญาณจากดิจิทัลเป็นอนาล็อก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.4 เส้นทางของข้อมูลทางดิจิทัล



รูปที่ 7.9 การแปลงข้อมูลแบบวันคอมพลิเมนต์และทวคอมพลิเมนต์

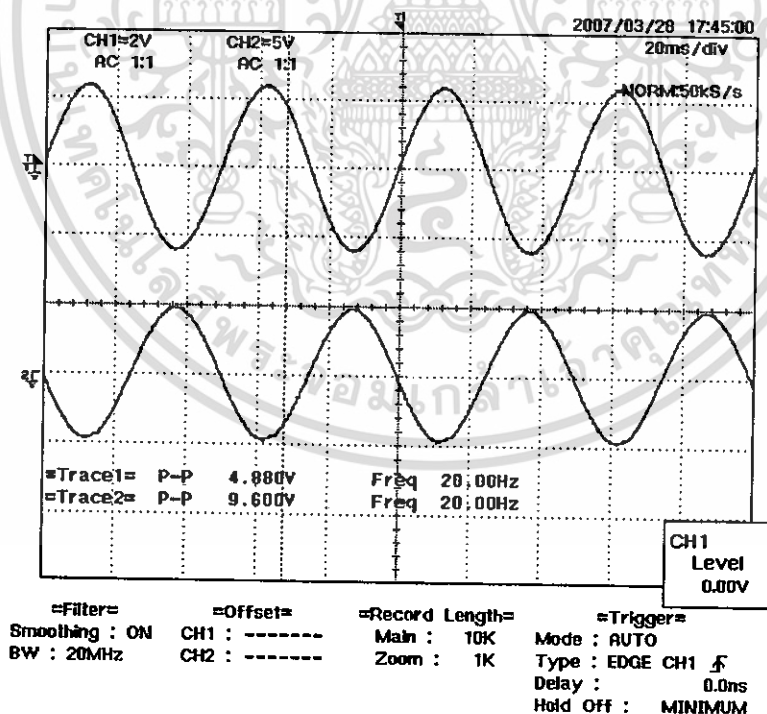
สัมประสิทธิ์ของตัวกรองมีทั้งเป็นบวกและเป็นลบ เราจึงจำเป็นต้องแปลงข้อมูลจากวันคอมพลิเมนต์เป็นทวคอมพลิเมนต์เพื่อให้คูณกับสัมประสิทธิ์ได้ สัญญาณที่ออกจากตัวแปลงสัญญาณจากอนาล็อกเป็นดิจิทัลเป็นวันคอมพลิเมนต์ โดยเราจะแปลงเป็นทวคอมพลิเมนต์ในด้านบวกเท่านั้น ทำให้สูญเสียข้อมูลไปหนึ่งบิต ก่อนที่จะคูณกับสัมประสิทธิ์แล้วกลายเป็นมีทั้งบวกและลบ แล้วจึงแปลงเป็นวันคอมพลิเมนต์เข้าตัวแปลงสัญญาณจากดิจิทัลเป็นอนาล็อกต่อไป

บทที่ 8

ผลการทดลอง

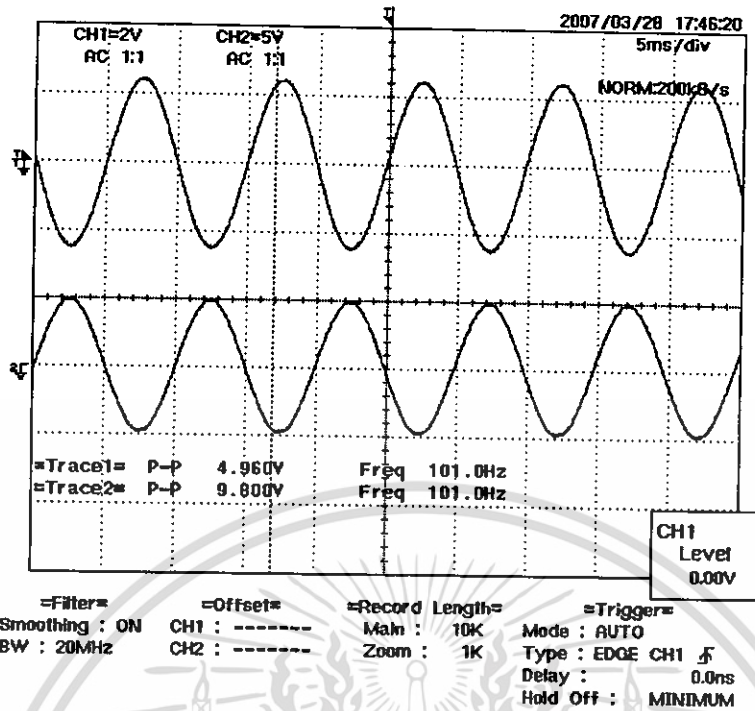
8.1 ตอนที่ 1 สัญญาณที่ผ่านตัวแปลงสัญญาณอนาล็อกเป็นดิจิทัลและตัวแปลงสัญญาณดิจิทัลเป็นอนาล็อก

ได้ทำการป้อนสัญญาณรูปขบวนขนาด 5 Vp-p ที่ความถี่ต่างๆ ให้วงจรแปลงสัญญาณอนาล็อกเป็นดิจิทัลและดิจิทัลเป็นอนาล็อก (ไม่มีตัวกรองสัญญาณ) เพื่อทดสอบวงจรแปลงสัญญาณว่าแปลงสัญญาณได้ถูกต้องเท่าใด รวมถึงความถี่ที่สามารถทำงานได้ ทำการสังเกตลักษณะสัญญาณของเอาต์พุตที่เปลี่ยนไปโดยให้สัญญาณที่ CH 1 แสดงสัญญาณอินพุต และให้สัญญาณที่ CH 2 แสดงสัญญาณเอาต์พุตโดยขนาดของสัญญาณเอาต์พุตจะอยู่ที่ 10 Vp-p จะได้ลักษณะสัญญาณดังรูปที่ 8.1 ถึง 8.9

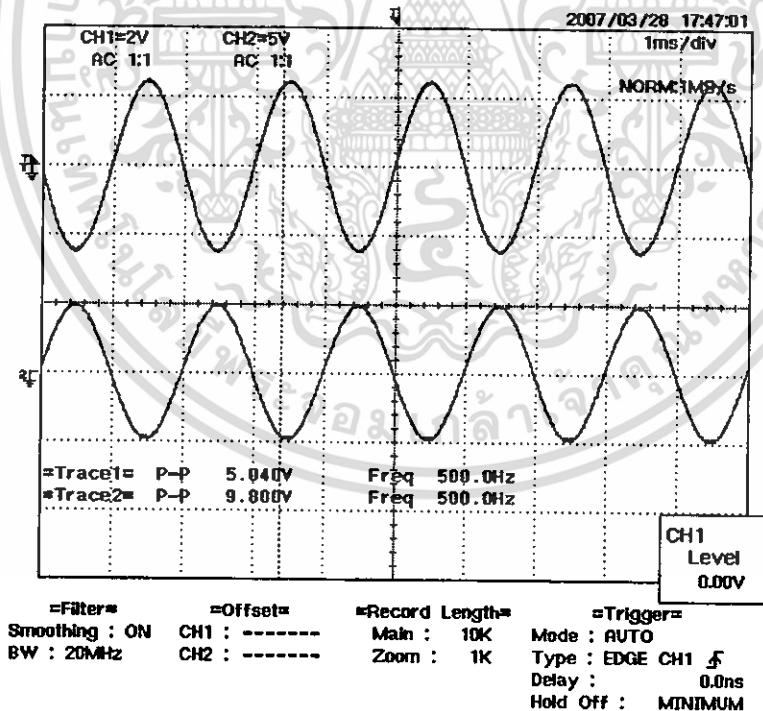


รูปที่ 8.1 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 20 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

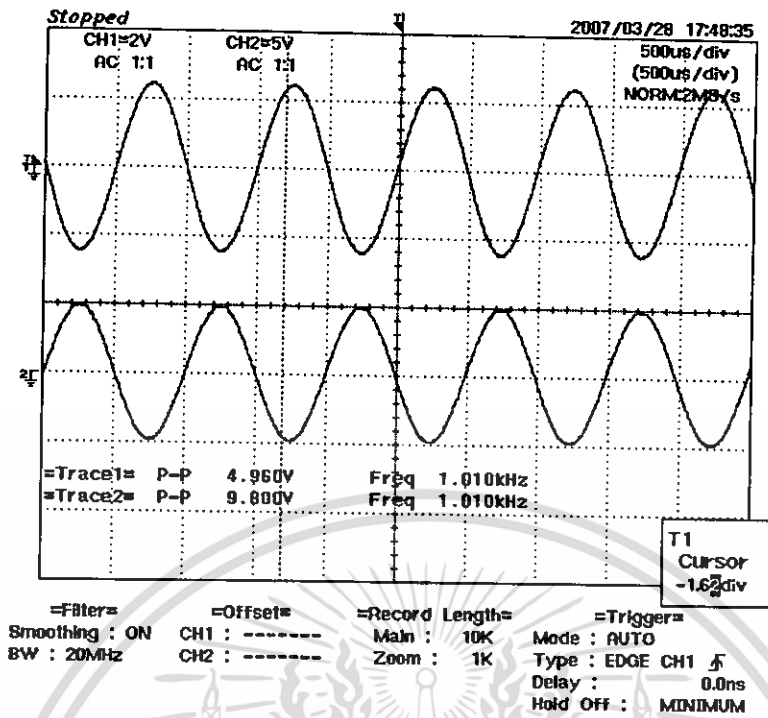


รูปที่ 8.2 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 100 Hz

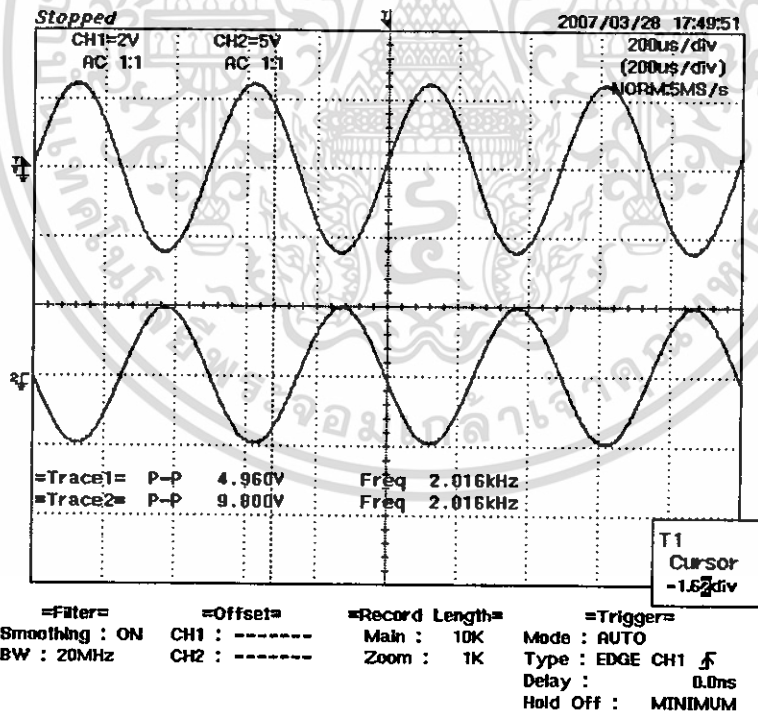


รูปที่ 8.3 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 500 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

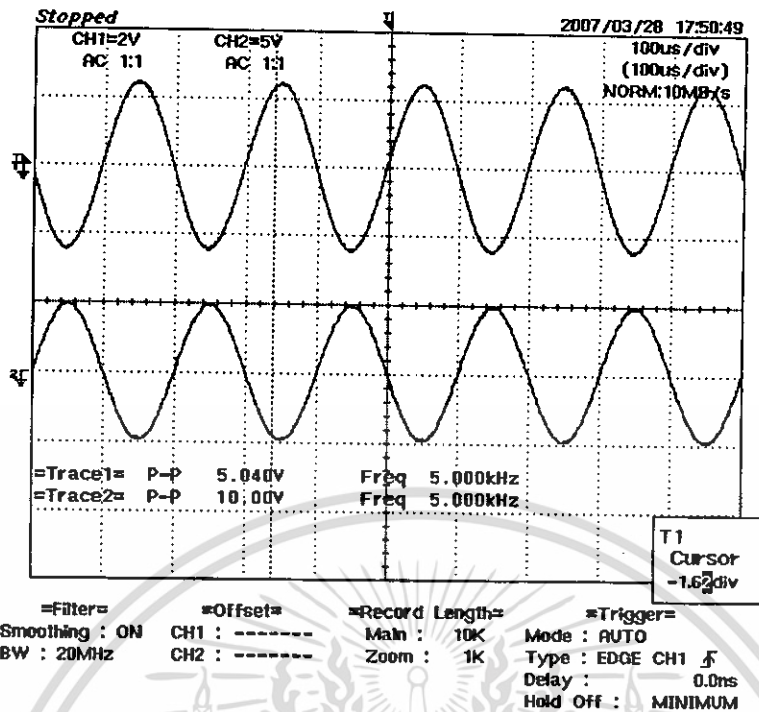


รูปที่ 8.4 ผลการทดลองคอนทที่ 1 เมื่อความถี่สัญญาณอินพุต 1 kHz

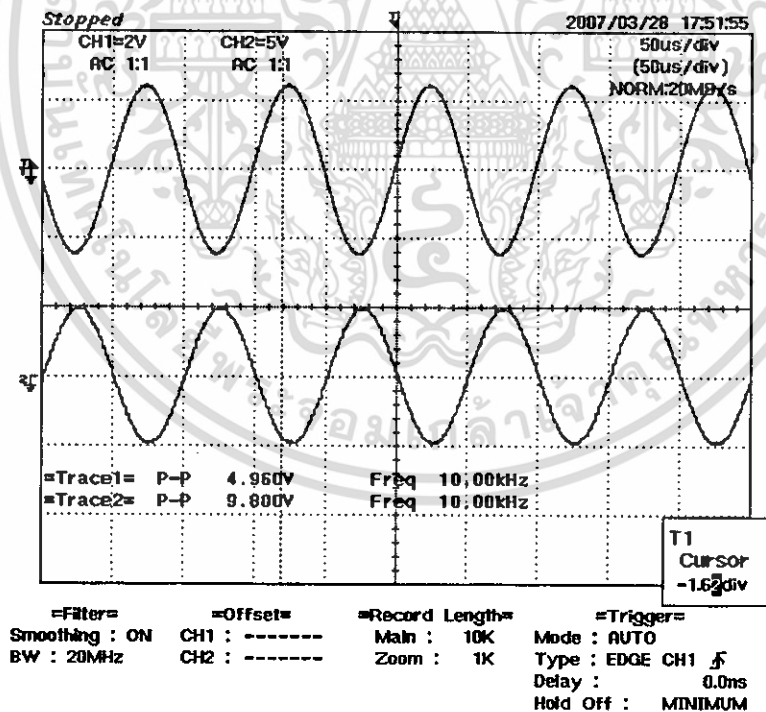


รูปที่ 8.5 ผลการทดลองคอนทที่ 1 เมื่อความถี่สัญญาณอินพุต 2 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

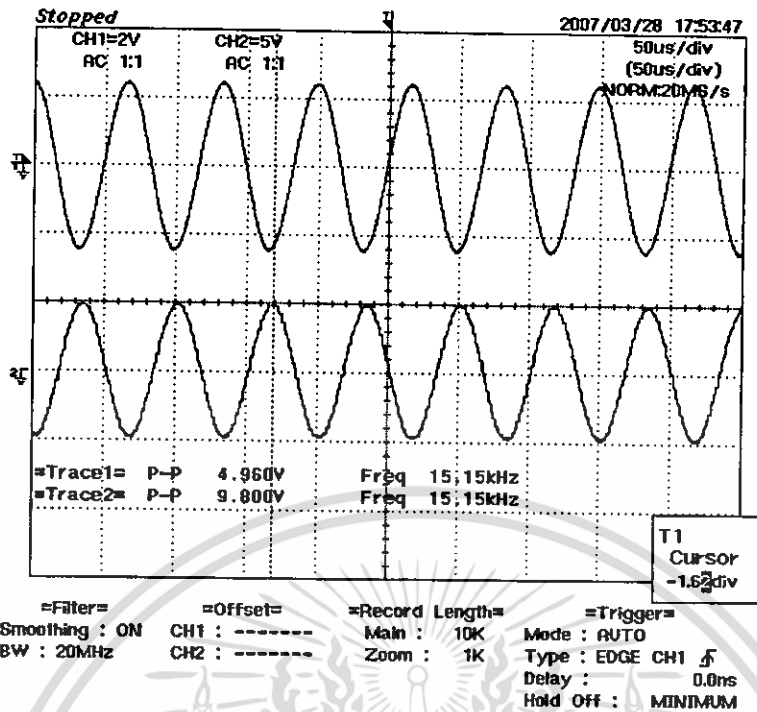


รูปที่ 8.6 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 5 kHz

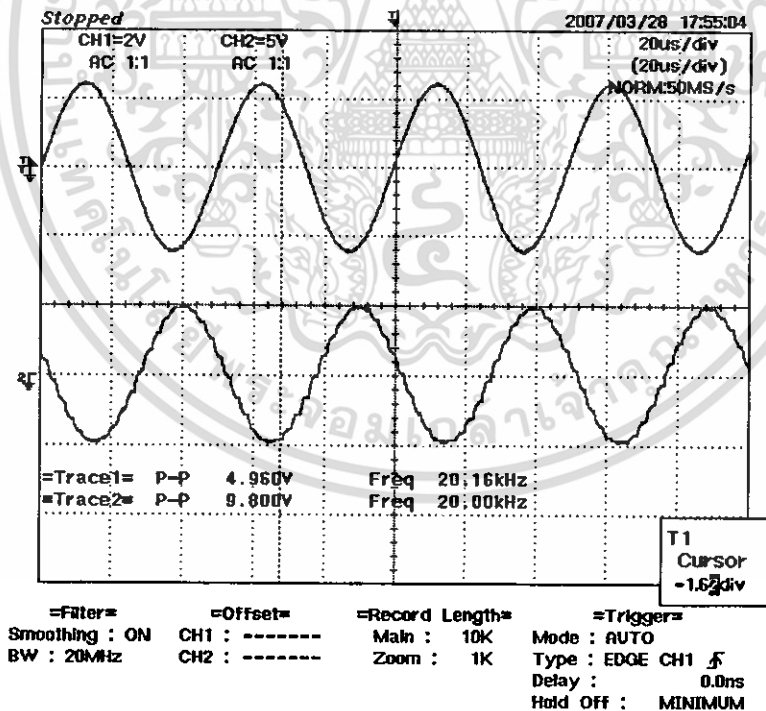


รูปที่ 8.7 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 10 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.8 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 15 kHz

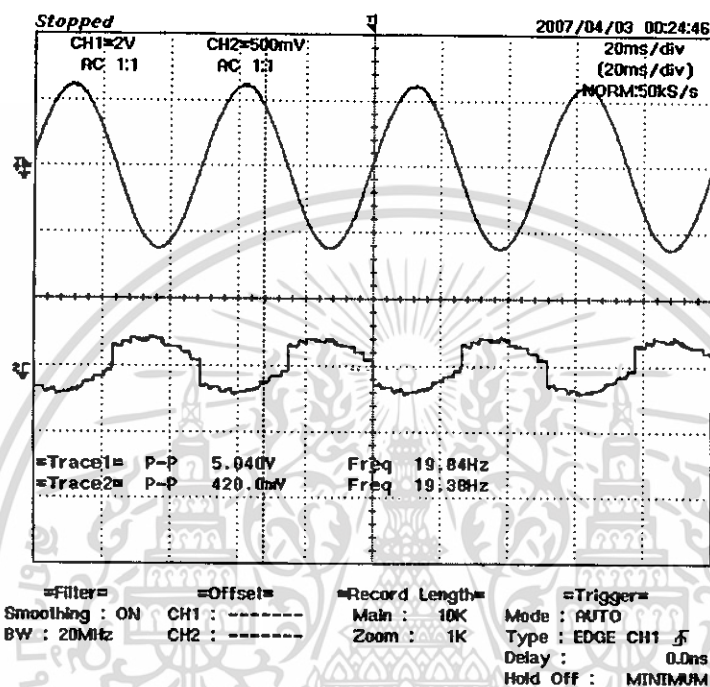


รูปที่ 8.9 ผลการทดลองตอนที่ 1 เมื่อความถี่สัญญาณอินพุต 20 kHz

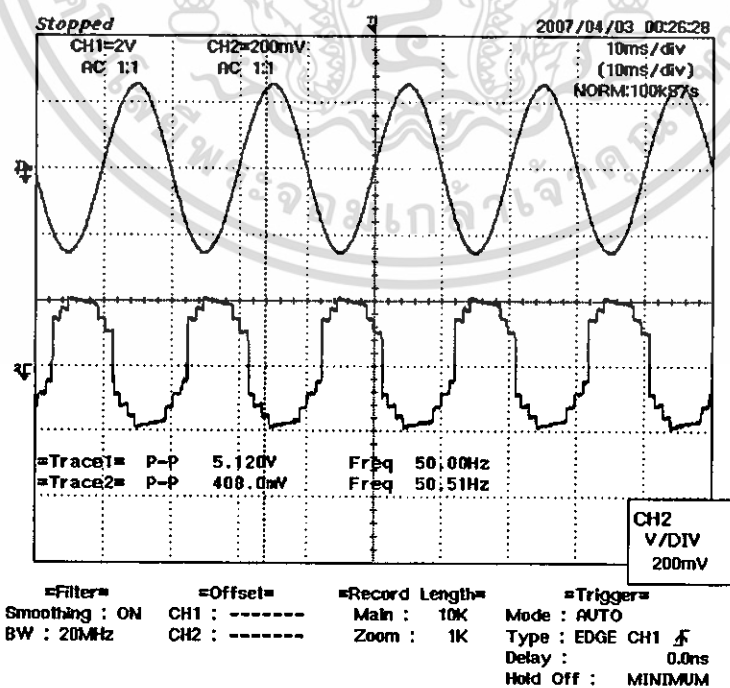
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.2 ตอนที่ 2 สัญญาณที่ผ่านตัวกรองความถี่แบบแถบผ่าน

หลังจากที่ได้ทำการสร้างตัวกรองสัญญาณและใส่ลงในซอฟต์แวร์พีซี ทำการป้อนสัญญาณที่มีความถี่ต่างๆมีขนาด 5 Vp-p ทำการสังเกตขนาดสัญญาณเอาต์พุต ได้ผลการทดลองดังรูปที่ 8.10 ถึง 8.25

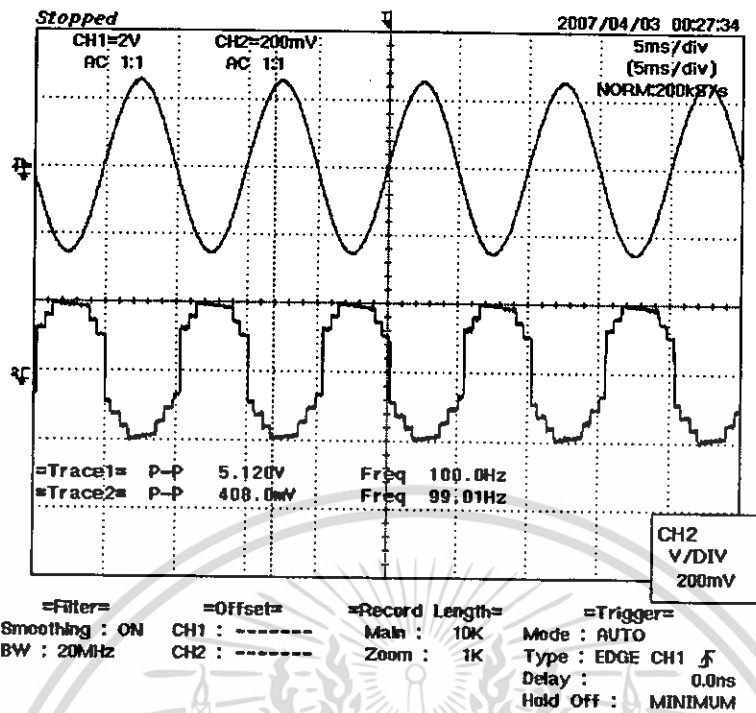


รูปที่ 8.10 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 20 Hz

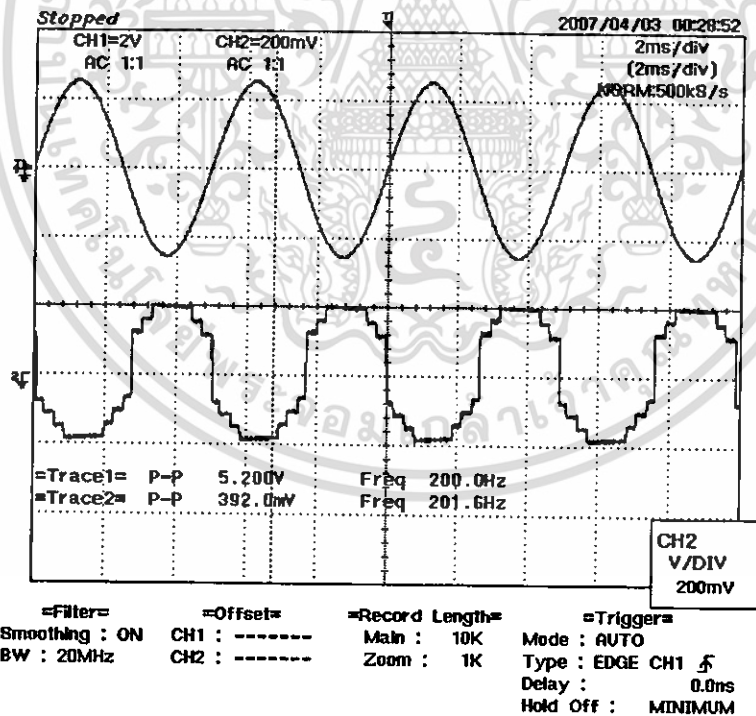


รูปที่ 8.11 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 50 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

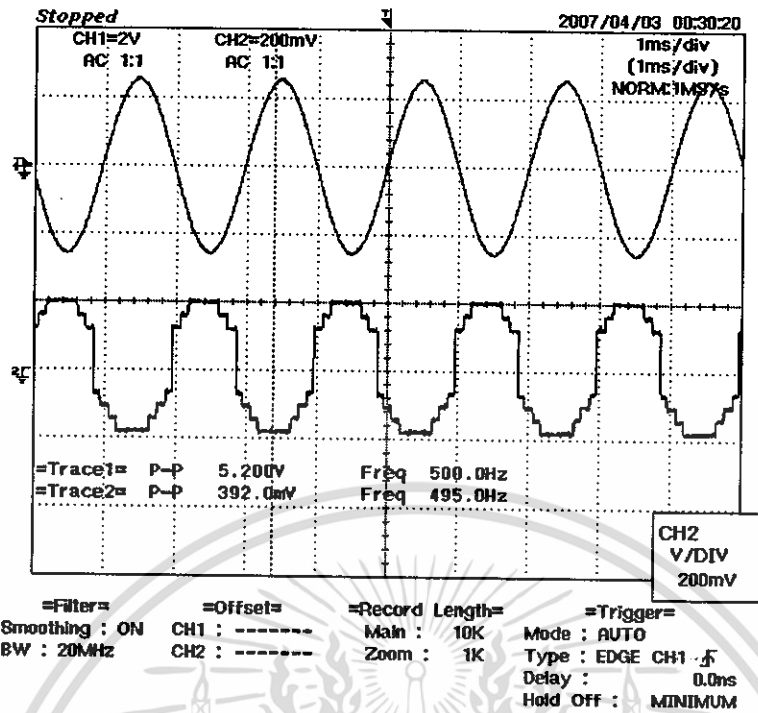


รูปที่ 8.12 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 100 Hz

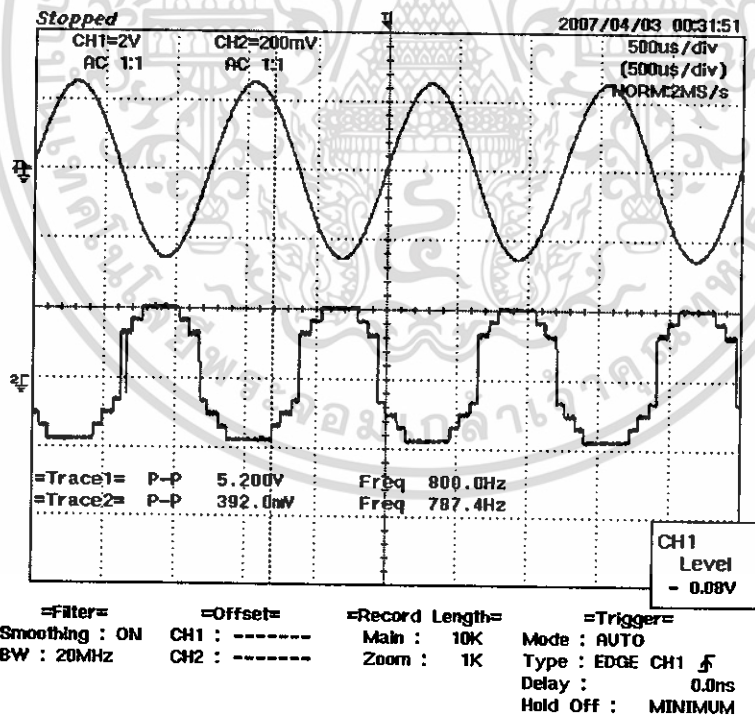


รูปที่ 8.13 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 200 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

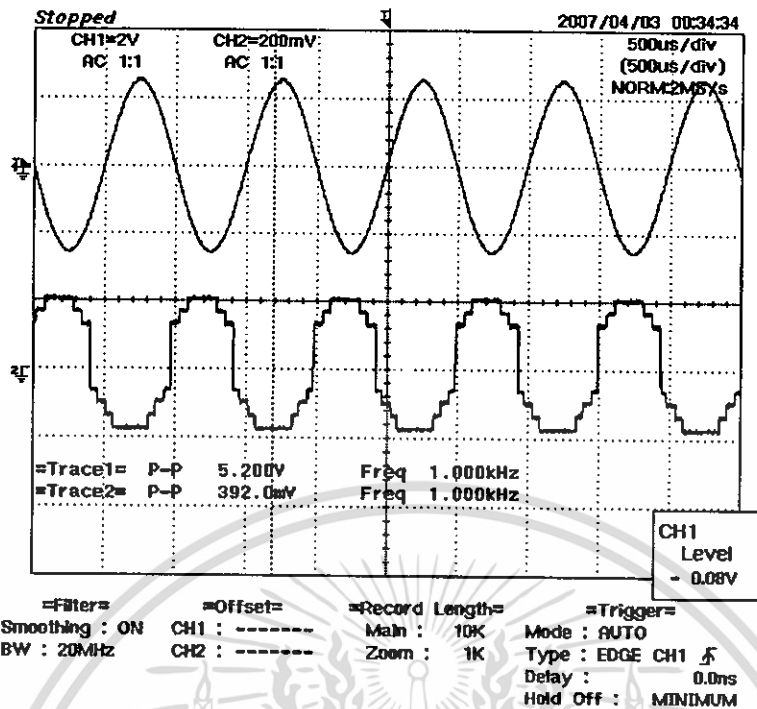


รูปที่ 8.14 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 500 Hz

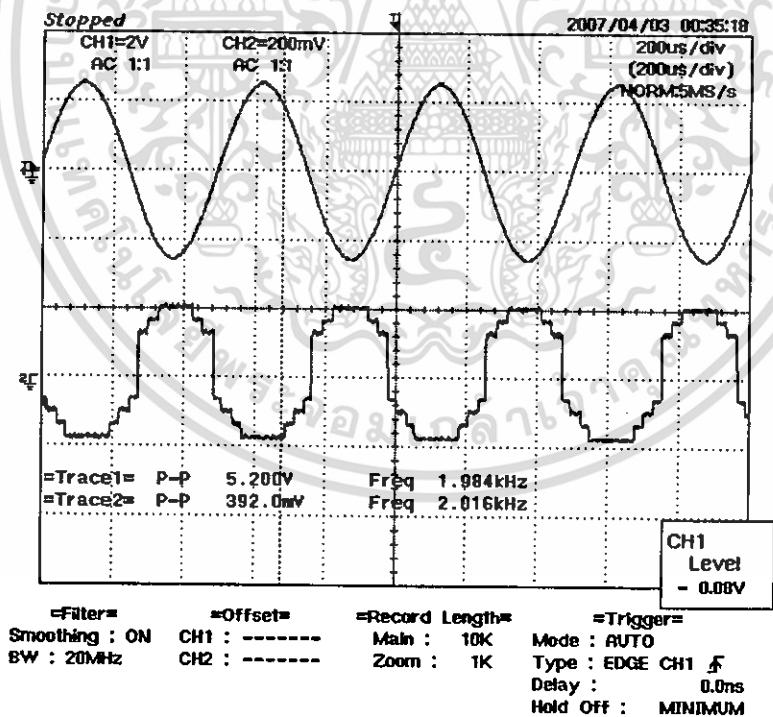


รูปที่ 8.15 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 800 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

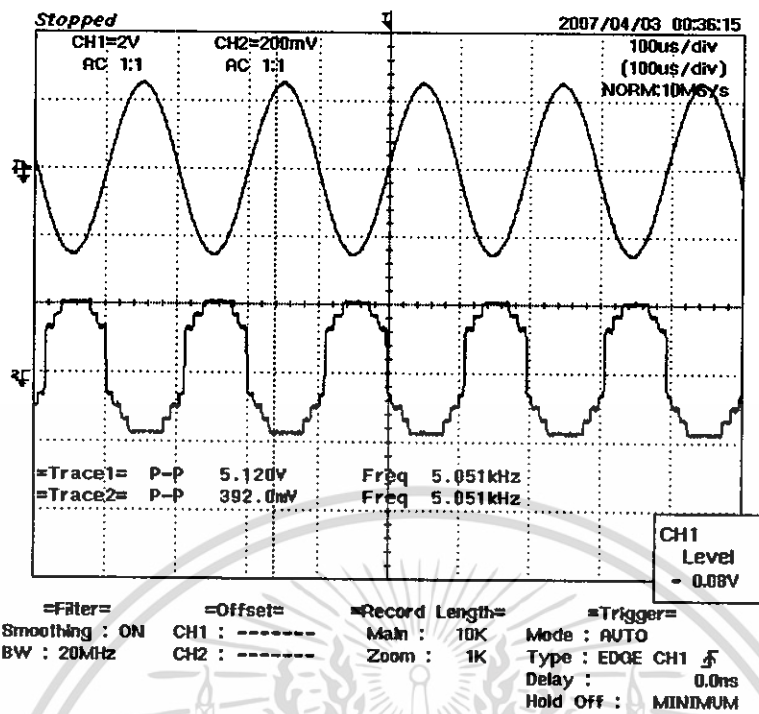


รูปที่ 8.16 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 1 kHz

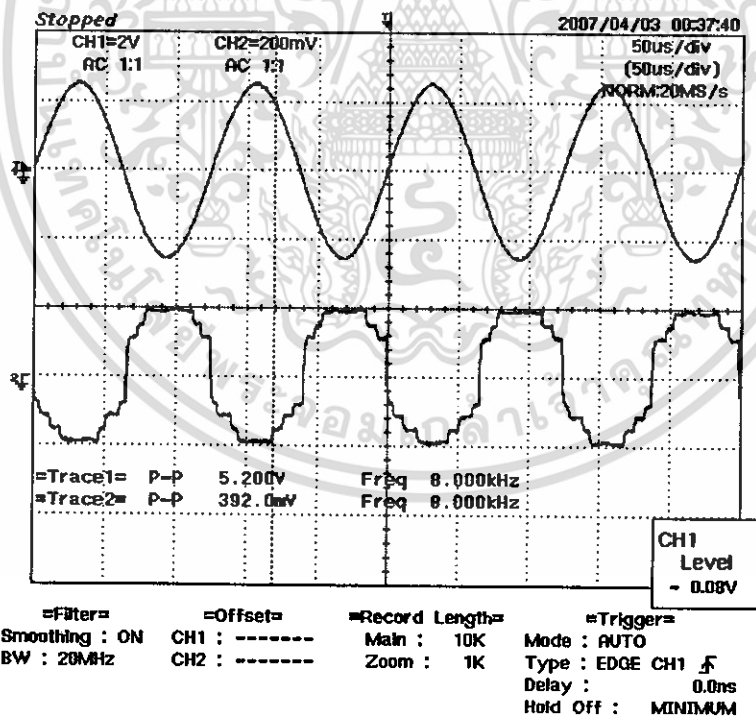


รูปที่ 8.17 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 2 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

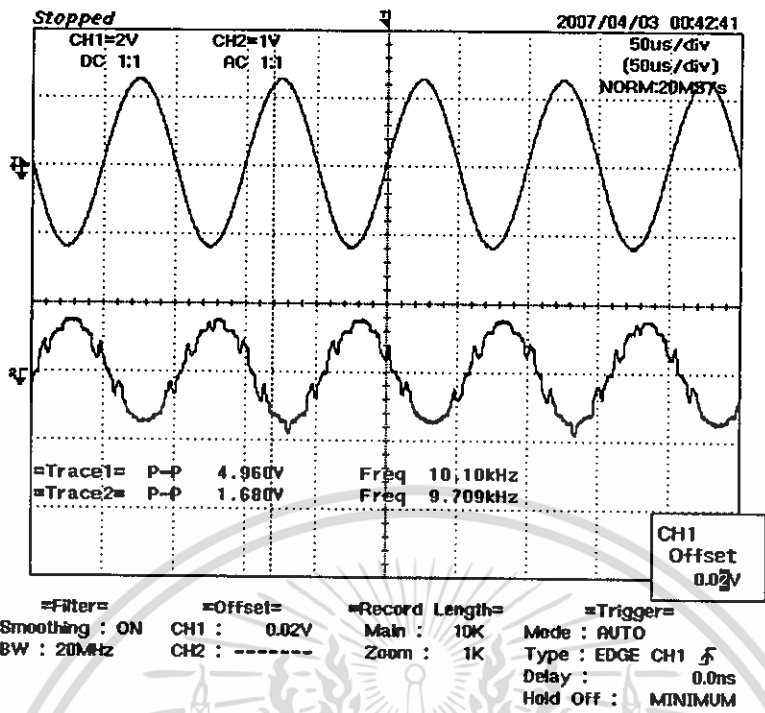


รูปที่ 8.18 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 5 kHz

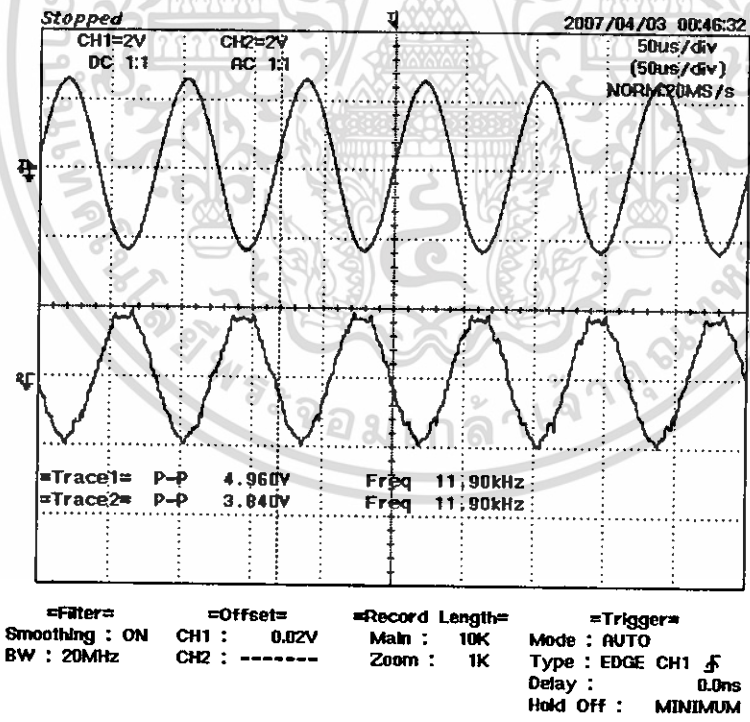


รูปที่ 8.19 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 8 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

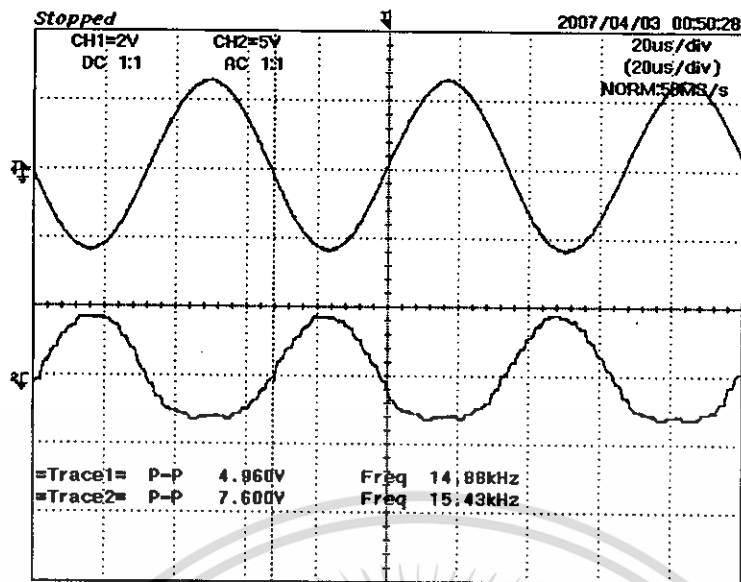


รูปที่ 8.20 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 10 kHz

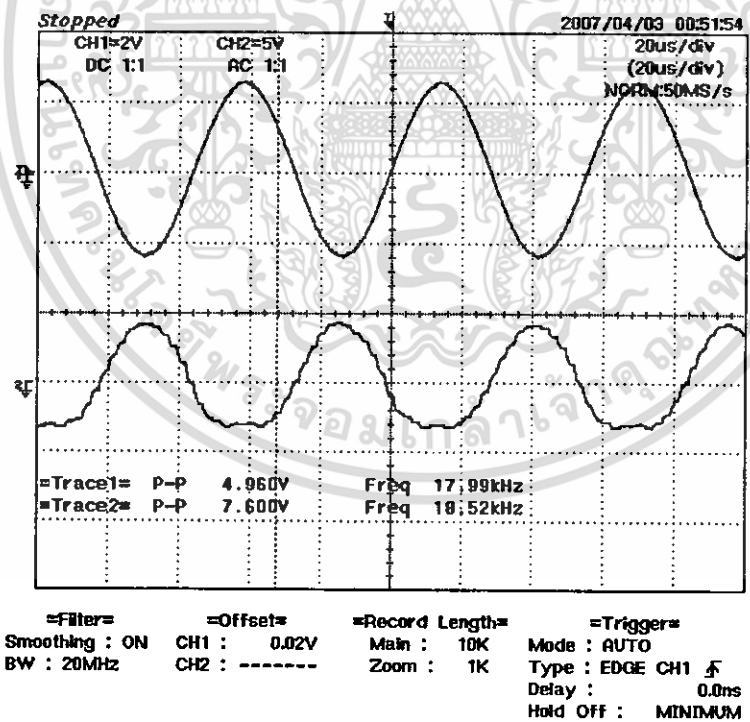


รูปที่ 8.21 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 12 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

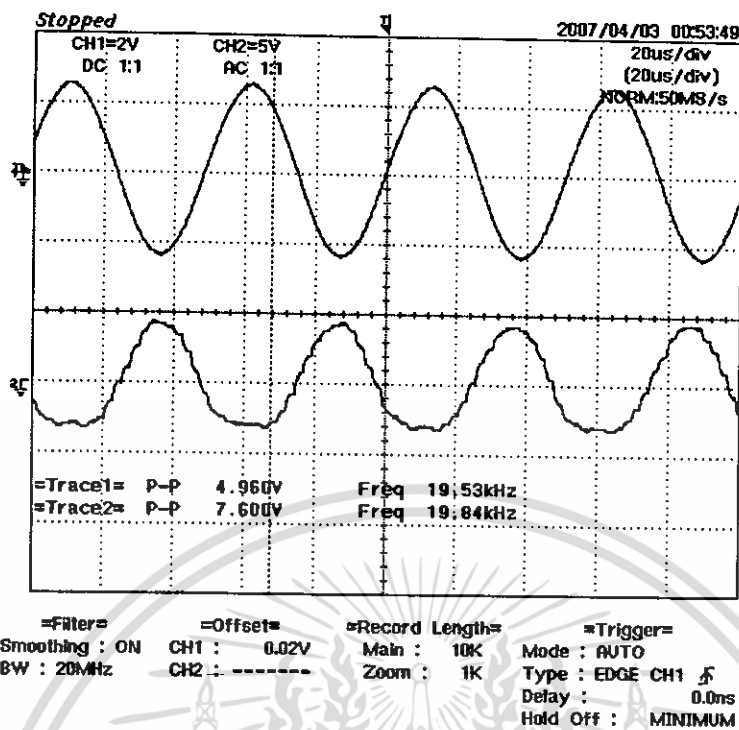


รูปที่ 8.22 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 15 kHz

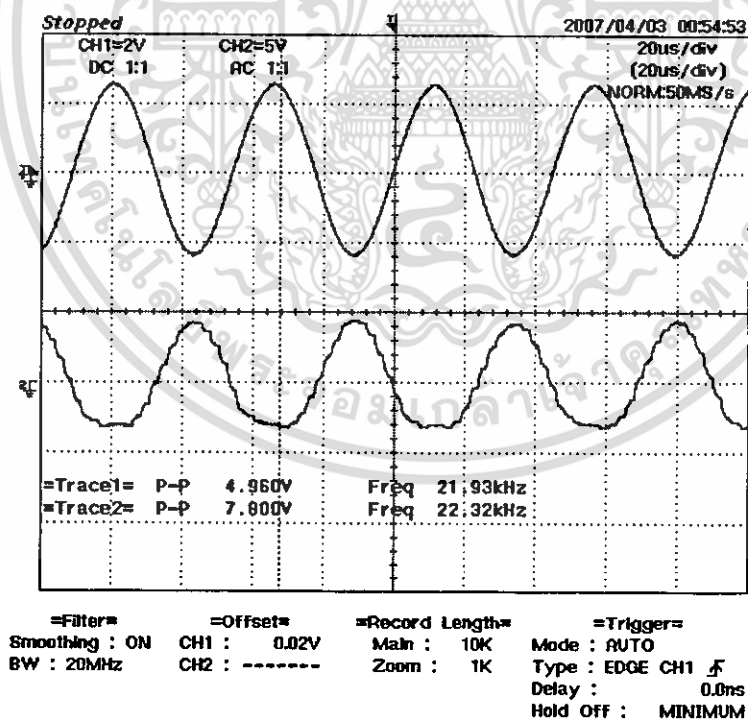


รูปที่ 8.23 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุต 18 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.24 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุท 20 kHz



รูปที่ 8.25 ผลการทดลองตอนที่ 2 เมื่อความถี่สัญญาณอินพุท 22 kHz

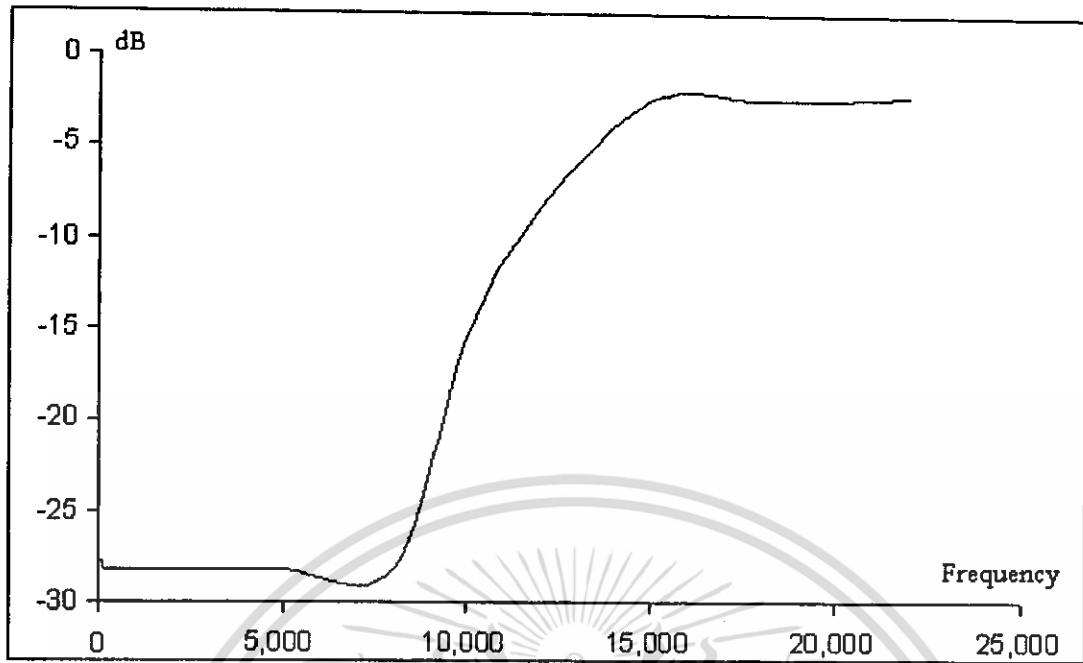
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3 ตอนที่ 3 กราฟแสดงผลการตอบสนองทางความถี่

นำขนาดของสัญญาณเอาต์พุตจากตอนที่ 2 รูปที่ 8.10 ถึง 8.25 มาใส่ลงในตารางที่ 8.1 และนำค่าจากตารางมาสร้างกราฟ ได้กราฟแสดงผลการตอบสนองทางความถี่ตามรูปที่ 8.26

ตารางที่ 8.1 ผลการทดลอง

สัญญาณอินพุตขนาด 5 V _{p-p} ที่ความถี่ต่างๆ	ขนาดของสัญญาณเอาต์พุต
20 Hz	420 mV
50 Hz	408 mV
100 Hz	408 mV
200 Hz	392 mV
500 Hz	392 mV
800 Hz	392 mV
1 kHz	392 mV
2 kHz	392 mV
5 kHz	392 mV
8 kHz	392 mV
10 kHz	1.68 V
12 kHz	3.84 V
15 kHz	7.6 V
18 kHz	7.6 V
20 kHz	7.6 V
22 kHz	7.8 V



รูปที่ 8.26 ผลการตอบสนองทางความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 9

บทสรุป

9.1 บทสรุป

โครงการนี้เป็นการสร้างตัวกรองความถี่ด้วยชิพ FPGA โดยใช้โปรแกรม Xilinx และ ภาษา VHDL ในการออกแบบ และ ใช้โปรแกรม Matlab ในการหาค่าสัมประสิทธิ์ของตัวกรอง ในการสร้างตัวกรองได้แบ่งออกเป็นส่วนใหญ่ๆ คือ วงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล ตัวแยกสัญญาณ ตัวคูณสัญญาณ ตัวรวมสัญญาณ และตัวแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก

จากการทดลองสร้างตัวกรองความถี่ด้วยชิพ FPGA เพื่อใช้ในงานอีควอไลเซอร์พบว่าสามารถทำได้แต่ต้องใช้เวลาในการศึกษาและทรัพยากรภายในชิพจำนวนมาก เพื่อให้ตัวกรองความถี่มีคุณภาพพอที่จะใช้งานได้

เมื่อเปรียบเทียบกราฟการตอบสนองทางความถี่ระหว่างผลการจำลองจากโปรแกรม Matlab กับ ผลการทดลองในบทที่ 8 พบว่ามีความใกล้เคียงกันแต่ยังคงมีความคลาดเคลื่อนสูงอยู่เนื่องจากผลของการปิดเศษข้อมูลทางดิจิทัลและความคลื่อนทางเวลาของลำดับการประมวลผลไม่ตรงกันส่งผลให้รูปสัญญาณเอาต์พุตผิดจากสัญญาณอินพุต

ในการใช้งานจริงเมื่อใช้วงจรกรองความถี่ต้องใช้ลำดับสูงมากเพื่อให้ตัวกรองมีคุณภาพที่ดี แต่ส่งผลให้ต้องใช้ทรัพยากรภายในชิพเพิ่มขึ้น ดังนั้นจึงต้องศึกษาหาเทคนิคที่มีประสิทธิภาพดีกว่านี้เพื่อให้ได้ตัวกรองที่มีประสิทธิภาพและใช้ทรัพยากรภายในชิพให้น้อยลงในฮาร์ดแวร์จริง

9.2 ปัญหาในโครงการ

- ในการใช้งานแบบเวลาจริงต้องคำนึงถึงลำดับการทำงานเป็นเวลา ซึ่งทำได้ยาก
- การใช้ตัวกรองลำดับสูงต้องใช้ทรัพยากรภายในชิพจำนวนมากเนื่องจากมีการประมวลผลทางสัญญาณมากขึ้น
- มีความคลาดเคลื่อนจากการปิดเศษของดิจิทัล โดยเฉพาะการเปลี่ยนค่าสัมประสิทธิ์เป็นเลขฐานสอง

9.3 แนวทางการแก้ปัญหา

- ปรับปรุงโปรแกรมในส่วนของ Timing Diagram ให้ดีขึ้นเพื่อลดความคลาดเคลื่อน
- ศึกษาหาวิธีการที่มีประสิทธิภาพเพื่อลดการใช้ทรัพยากรภายในชิพ FPGA
- ใช้ตัวแปลงสัญญาณอนาล็อกเป็นดิจิทัลที่มีความสามารถในการทำงานที่ละเอียดมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปเผยแพร่โดยไม่เสียค่าใช้จ่าย
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. ศ. ดร. วัลลภ สุระกำพลธร , เอกสารประกอบการเรียนการสอนวิชาการประมวลสัญญาณเชิงเลข
2. สุกัญญา แพระสมบุรณ์, “การออกแบบตัวกรองเชิงเลข FIR หลายเอาต์พุตแบบเวลาจริง”, วิทยานิพนธ์ วศ.ม. โทคมណาคม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2549
3. ชานนท์ ศรีน้อย, อรรถกร ฉายรัมย์, “อีควอไลเซอร์”, วิทยานิพนธ์ วศ.บ. โทคมណาคม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2546
4. รศ.ดร. สมศักดิ์ ชุ่มช่วย , “การประมวลผลสัญญาณเชิงเลข”, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2545
5. รศ.ดร. มนัส สัจจวรศิลป์, วรรัตน์ ภัทรอมรกุล, “คู่มือการใช้งาน Matlab ฉบับสมบูรณ์” Info press, 2543
6. Vinay K. Ingle, John G. Proakis, “Digital Signal Processing Using Matlab”, Northeastern University, PWS Publishing Company, 1997
7. ภาษา VHDL สำหรับการออกแบบวงจรดิจิทัล, ชำนาญ ปัญญาใส วัชรกรหนุทอง

ADC0820 8-Bit High Speed μ P Compatible A/D Converter with Track/Hold Function

General Description

By using a half-flash conversion technique, the 8-bit ADC0820 CMOS A/D offers a 1.5 μ s conversion time and dissipates only 75 mW of power. The half-flash technique consists of 32 comparators, a most significant 4-bit ADC and a least significant 4-bit ADC.

The input to the ADC0820 is tracked and held by the input sampling circuitry eliminating the need for an external sample-and-hold for signals moving at less than 100 mV/ μ s.

For ease of interface to microprocessors, the ADC0820 has been designed to appear as a memory location or I/O port without the need for external interfacing logic.

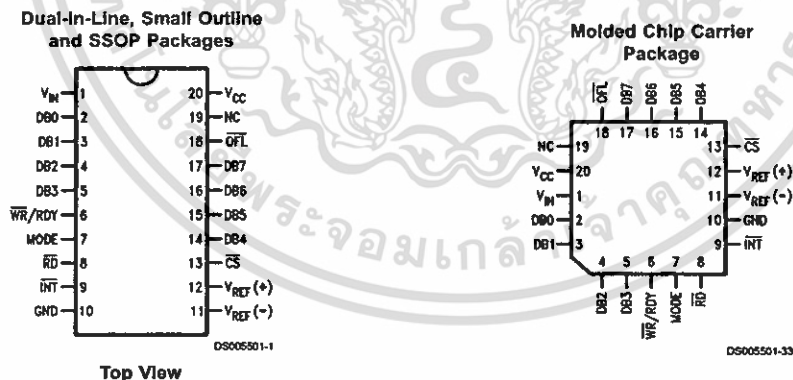
Key Specifications

■ Resolution	8 Bits
■ Conversion Time	2.5 μ s Max (RD Mode) 1.5 μ s Max (WR-RD Mode)
■ Low Power	75 mW Max
■ Total Unadjusted Error	$\pm 1/2$ LSB and ± 1 LSB

Features

- Built-in track-and-hold function
- No missing codes
- No external clocking
- Single supply—5 V_{DC}
- Easy interface to all microprocessors, or operates stand-alone
- Latched TRI-STATE® output
- Logic inputs and outputs meet both MOS and T²L voltage level specifications
- Operates ratiometrically or with any reference value equal to or less than V_{CC}
- 0V to 5V analog input voltage range with single 5V supply
- No zero or full-scale adjust required
- Overflow output available for cascading
- 0.3" standard width 20-pin DIP
- 20-pin molded chip carrier package
- 20-pin small outline package
- 20-pin shrink small outline package (SSOP)

Connection and Functional Diagrams



TRI-STATE® is a registered trademark of National Semiconductor Corporation.

Connection and Functional Diagrams (Continued)

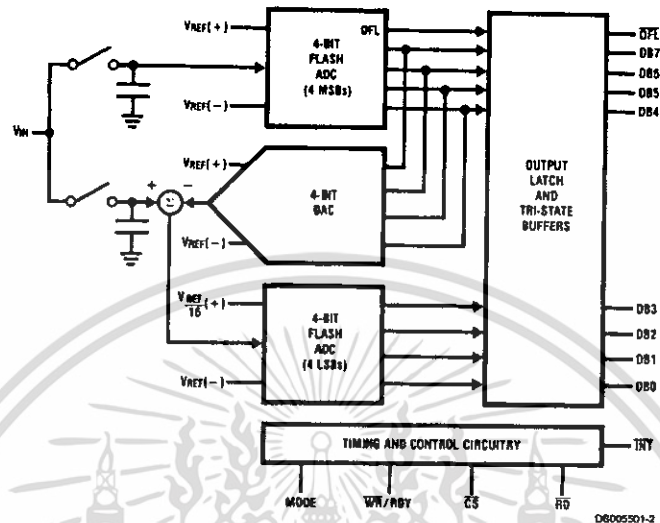


FIGURE 1.

Ordering Information

Part Number	Total Unadjusted Error	Package	Temperature Range
ADC0820BCV	$\pm \frac{1}{2}$ LSB	V20A — Molded Chip Carrier	0°C to +70°C
ADC0820BCWM		M20B — Wide Body Small Outline	0°C to +70°C
ADC0820BCN		N20A — Molded DIP	0°C to +70°C
ADC0820CCJ	± 1 LSB	J20A — Cerdip	-40°C to +85°C
ADC0820CCWM		M20B — Wide Body Small Outline	0°C to +70°C
ADC0820CIWM		M20B — Wide Body Small Outline	-40°C to +85°C
ADC0820CCN		N20A — Molded DIP	0°C to +70°C

Absolute Maximum Ratings (Notes 1, 2)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Supply Voltage (V_{CC})	10V
Logic Control Inputs	-0.2V to V_{CC} +0.2V
Voltage at Other Inputs and Output	-0.2V to V_{CC} +0.2V
Storage Temperature Range	-65°C to +150°C
Package Dissipation at $T_A = 25^\circ\text{C}$	875 mW
Input Current at Any Pin (Note 5)	1 mA
Package Input Current (Note 5)	4 mA
ESD Susceptibility (Note 9)	1200V
Lead Temp. (Soldering, 10 sec.)	260°C
Dual-In-Line Package (plastic)	260°C

Dual-In-Line Package (ceramic)	300°C
Surface Mount Package	
Vapor Phase (60 sec.)	215°C
Infrared (15 sec.)	220°C

Operating Ratings (Notes 1, 2)

Temperature Range	$T_{MIN} \leq T_A \leq T_{MAX}$
ADC0820CCJ	-40°C $\leq T_A \leq$ +85°C
ADC0820CIWM	-40°C $\leq T_A \leq$ +85°C
ADC0820BCN, ADC0820CCN	0°C $\leq T_A \leq$ 70°C
ADC0820BCV	0°C $\leq T_A \leq$ 70°C
ADC0820BCWM, ADC0820CCWM	0°C $\leq T_A \leq$ 70°C
V_{CC} Range	4.5V to 8V

Converter Characteristics

The following specifications apply for RD mode (pin 7=0), $V_{CC}=5V$, $V_{REF(+)}=5V$, and $V_{REF(-)}=GND$ unless otherwise specified. **Boldface limits apply from T_{MIN} to T_{MAX} ; all other limits $T_A=T_J=25^\circ\text{C}$.**

Parameter	Conditions	ADC0820CCJ			ADC0820BCN, ADC0820CCN ADC0820BCV, ADC0820BCWM ADC0820CCWM, ADC0820CIWM			Limit Units
		Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	
Resolution			8		8	8	Bits	
Total Unadjusted Error (Note 3)	ADC0820BCN, BCWM ADC0820CCJ ADC0820CCN, CCWM, CIWM, ADC0820CCMSA		± 1		$\pm 1/2$ ± 1 ± 1	$\pm 1/2$ ± 1 ± 1	LSB LSB LSB	
Minimum Reference Resistance		2.3	1.00		2.3	1.2	k Ω	
Maximum Reference Resistance		2.3	6		2.3	5.3	k Ω	
Maximum $V_{REF(+)}$ Input Voltage			V_{CC}		V_{CC}	V_{CC}	V	
Minimum $V_{REF(-)}$ Input Voltage			GND		GND	GND	V	
Minimum $V_{REF(+)}$ Input Voltage			$V_{REF(-)}$		$V_{REF(-)}$	$V_{REF(-)}$	V	
Maximum $V_{REF(-)}$ Input Voltage			$V_{REF(+)}$		$V_{REF(+)}$	$V_{REF(+)}$	V	
Maximum V_{IN} Input Voltage			$V_{CC}+0.1$		$V_{CC}+0.1$	$V_{CC}+0.1$	V	
Minimum V_{IN} Input Voltage			GND-0.1		GND-0.1	GND-0.1	V	
Maximum Analog Input Leakage Current	$\overline{CS} = V_{CC}$ $V_{IN} = V_{CC}$ $V_{IN} = GND$		3 -3		0.3 -0.3	3 -3	μA μA	
Power Supply Sensitivity	$V_{CC} = 5V \pm 5\%$	$\pm 1/16$	$\pm 1/4$		$\pm 1/16$	$\pm 1/4$	LSB	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC Electrical Characteristics

The following specifications apply for $V_{CC}=5V$, unless otherwise specified. **Boldface limits apply from T_{MIN} to T_{MAX}** ; all other limits $T_A=T_J=25^{\circ}C$.

Parameter	Conditions		ADC0820CCJ			ADC0820BCN, ADC0820CCN ADC0820BCV, ADC0820BCWM ADC0820CCWM, ADC0820CIWM			Limit Units
			Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	
			$V_{IN(1)}$, Logical "1" Input Voltage	$V_{CC}=5.25V$	\overline{CS} , \overline{WR} , \overline{RD} Mode		2.0		
$V_{IN(0)}$, Logical "0" Input Voltage	$V_{CC}=4.75V$	\overline{CS} , \overline{WR} , \overline{RD} Mode		3.5			3.5	3.5	V
$V_{IN(1)}$, Logical "1" Input Current	$V_{IN(1)}=5V$; \overline{CS} , \overline{RD} $V_{IN(1)}=5V$; \overline{WR} $V_{IN(1)}=5V$; Mode		0.005	1			0.005	1	μA
$V_{IN(0)}$, Logical "0" Input Current	$V_{IN(0)}=0V$; \overline{CS} , \overline{RD} , \overline{WR} , Mode		0.1	3			0.1	0.3	μA
$V_{OUT(1)}$, Logical "1" Output Voltage	$V_{CC}=4.75V$, $I_{OUT}=-360 \mu A$; DB0–DB7, \overline{OFL} , \overline{INT} $V_{CC}=4.75V$, $I_{OUT}=-10 \mu A$; DB0–DB7, \overline{OFL} , \overline{INT}			50			50	170	μA
$V_{OUT(0)}$, Logical "0" Output Voltage	$V_{CC}=4.75V$, $I_{OUT}=1.6 mA$; DB0–DB7, \overline{OFL} , \overline{INT} , \overline{RDY}		-0.005	-1			-0.005	-1	μA
I_{OUT} , TRI-STATE Output Current	$V_{OUT}=5V$; DB0–DB7, \overline{RDY} $V_{OUT}=0V$; DB0–DB7, \overline{RDY}		0.005	200			0.005	200	μA
I_{SOURCE} , Output Source Current	$V_{OUT}=0V$; DB0–DB7, \overline{OFL} , \overline{INT}		0.4				0.34	0.4	V
I_{SNK} , Output Sink Current	$V_{OUT}=5V$; DB0–DB7, \overline{OFL} , \overline{INT} , \overline{RDY}		0.1	3			0.1	0.3	μA
I_{CC} , Supply Current	$\overline{CS}=\overline{WR}=\overline{RD}=0$		-0.1	-3			-0.1	-0.3	μA
			-12	-6			-12	-7.2	mA
			-9	-4.0			-9	-5.3	mA
			14	7			14	8.4	mA
			7.5	15			7.5	13	mA

AC Electrical Characteristics

The following specifications apply for $V_{CC}=5V$, $t_r=t_f=20 ns$, $V_{REF(+)}=5V$, $V_{REF(-)}=0V$ and $T_A=25^{\circ}C$ unless otherwise specified.

Parameter	Conditions	Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	Units
t_{CRD} , Conversion Time for RD Mode	Pin 7 = 0, Figure 2	1.6		2.5	μs
t_{ACC0} , Access Time (Delay from Falling Edge of \overline{RD} to Output Valid)	Pin 7 = 0, Figure 2	$t_{CRD}+20$		$t_{CRD}+50$	ns
t_{CWR-RD} , Conversion Time for WR-RD Mode	Pin 7 = V_{CC} ; $t_{WR} = 600 ns$, $t_{RD} = 600 ns$; Figures 3, 4			1.52	μs
t_{WR} , Write Time	Min		600		ns
	Max		50		μs
t_{RD} , Read Time	Min		600		ns
t_{ACC1} , Access Time (Delay from Falling Edge of \overline{RD} to Output Valid)	Pin 7 = V_{CC} , $t_{RD} < t_r$; Figure 3 $C_L = 15 pF$	190		280	ns
	$C_L = 100 pF$	210		320	ns

AC Electrical Characteristics (Continued)

The following specifications apply for $V_{CC}=5V$, $t_r=t_f=20$ ns, $V_{REF(+)}=5V$, $V_{REF(-)}=0V$ and $T_A=25^\circ C$ unless otherwise specified.

Parameter	Conditions	Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	Units
t_{ACC2} , Access Time (Delay from Falling Edge of \overline{RD} to Output Valid)	Pin 7 = V_{CC} , $t_{RD} > t_i$; Figure 4 $C_L=15$ pF	70		120	ns
	$C_L=100$ pF	90		150	ns
t_{ACC3} , Access Time (Delay from Rising Edge of \overline{RDY} to Output Valid)	$R_{PULLUP} = 1k$ and $C_L = 15$ pF	30			ns
t_i , Internal Comparison Time	Pin 7 = V_{CC} ; Figures 4, 5 $C_L=50$ pF	800		1300	ns
t_{IH} , t_{OH} , TRI-STATE Control (Delay from Rising Edge of \overline{RD} to Hi-Z State)	$R_L=1k$, $C_L=10$ pF	100		200	ns
t_{INTL} , Delay from Rising Edge of \overline{WR} to Falling Edge of \overline{INT}	Pin 7 = V_{CC} , $C_L = 50$ pF $t_{RD} > t_i$; Figure 4	$t_{RD}+200$		t_i	ns
	$t_{RD} < t_i$; Figure 3			$t_{RD}+290$	ns
t_{INTH} , Delay from Rising Edge of \overline{RD} to Rising Edge of \overline{INT}	Figures 2, 3, 4 $C_L=50$ pF	125		225	ns
t_{INTHWR} , Delay from Rising Edge of \overline{WR} to Rising Edge of \overline{INT}	Figure 5, $C_L=50$ pF	175		270	ns
t_{RDY} , Delay from \overline{CS} to \overline{RDY}	Figure 2, $C_L=50$ pF, Pin 7 = 0	50		100	ns
t_{D} , Delay from \overline{INT} to Output Valid	Figure 5	20		50	ns
t_{RI} , Delay from \overline{RD} to \overline{INT}	Pin 7 = V_{CC} , $t_{RD} < t_i$ Figure 3	200		290	ns
t_p , Delay from End of Conversion to Next Conversion	Figures 2, 3, 4, 5 (Note 4) See Graph			500	ns
Slew Rate, Tracking		0.1			V/ μ s
C_{VIN} , Analog Input Capacitance		45			pF
C_{OUT} , Logic Output Capacitance		5			pF
C_{IN} , Logic Input Capacitance		5			pF

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its specified operating conditions.

Note 2: All voltages are measured with respect to the GND pin, unless otherwise specified.

Note 3: Total unadjusted error includes offset, full-scale, and linearity errors.

Note 4: Accuracy may degrade if t_{WR} or t_{RD} is shorter than the minimum value specified. See Accuracy vs t_{WR} and Accuracy vs t_{RD} graphs.

Note 5: When the input voltage (V_{IN}) at any pin exceeds the power supply rails ($V_{IN} < V^-$ or $V_{IN} > V^+$) the absolute value of current at that pin should be limited to 1 mA or less. The 4 mA package input current limits the number of pins that can exceed the power supply boundaries with a 1 mA current limit to four.

Note 6: Typicals are at 25°C and represent most likely parametric norm.

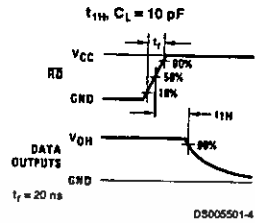
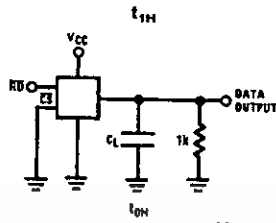
Note 7: Tested limits are guaranteed to National's AOQL (Average Outgoing Quality Level).

Note 8: Design limits are guaranteed but not 100% tested. These limits are not used to calculate outgoing quality levels.

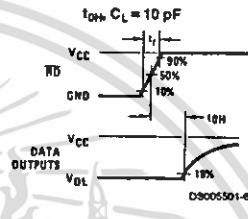
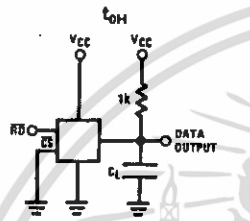
Note 9: Human body model, 100 pF discharged through a 1.5 k Ω resistor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRI-STATE Test Circuits and Waveforms

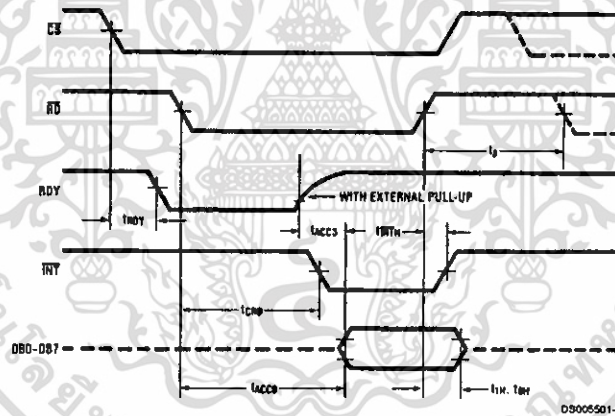


$t_L = 20 \text{ ns}$



$t_L = 20 \text{ ns}$

Timing Diagrams



Note: On power-up the state of $\overline{\text{INT}}$ can be high or low.

FIGURE 2. RD Mode (Pin 7 is Low)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Timing Diagrams (Continued)

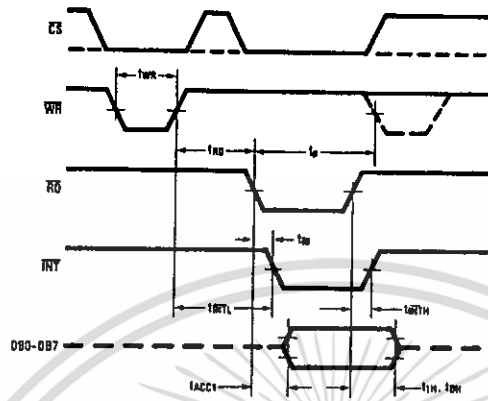


FIGURE 3. WR-RD Mode (Pin 7 is High and $t_{RD} < t_1$)

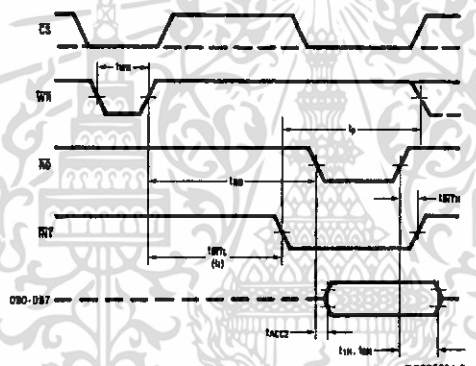


FIGURE 4. WR-RD Mode (Pin 7 is High and $t_{RD} > t_1$)

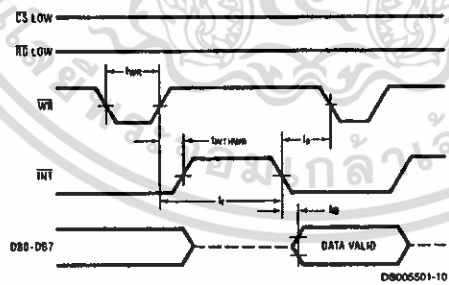
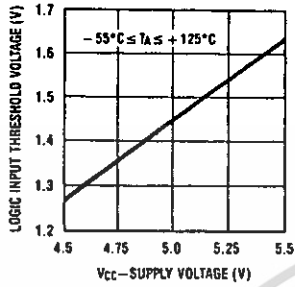


FIGURE 5. WR-RD Mode (Pin 7 is High) Stand-Alone Operation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

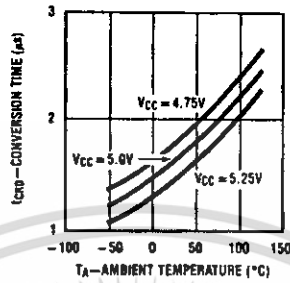
Typical Performance Characteristics

Logic Input Threshold Voltage vs Supply Voltage



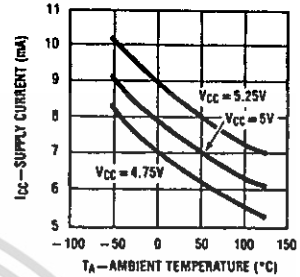
DS005501-34

Conversion Time (RD Mode) vs Temperature



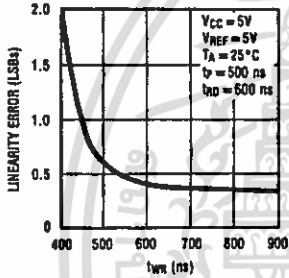
DS005501-35

Power Supply Current vs Temperature (not including reference ladder)



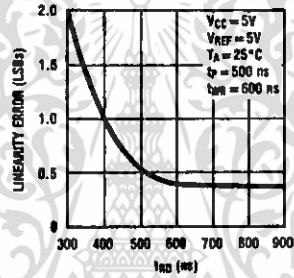
DS005501-36

Accuracy vs t_{WR}



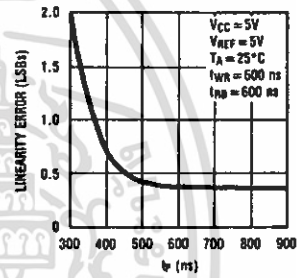
DS005501-37

Accuracy vs t_{RD}



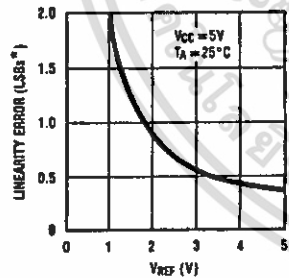
DS005501-38

Accuracy vs t_p



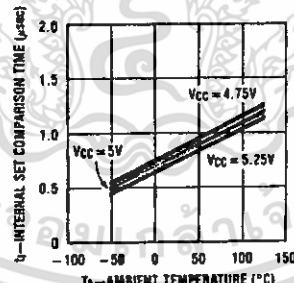
DS005501-39

Accuracy vs V_{REF}
[$V_{REF} = V_{REF(+)} - V_{REF(-)}$]



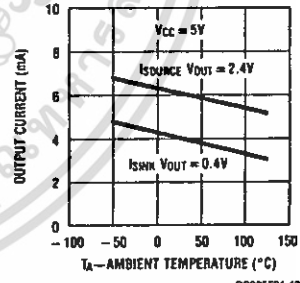
DS005501-40

t_i , Internal Time Delay vs Temperature



DS005501-41

Output Current vs Temperature



DS005501-42

$$*1 \text{ LSB} = \frac{V_{REF}}{256}$$

Description of Pin Functions

Pin	Name	Function
1	V_{IN}	Analog input; range $=GND \leq V_{IN} \leq V_{CC}$
2	DB0	TRI-STATE data output—bit 0 (LSB)
3	DB1	TRI-STATE data output—bit 1
4	DB2	TRI-STATE data output—bit 2
5	DB3	TRI-STATE data output—bit 3
6	\overline{WR} /RDY	WR-RD Mode \overline{WR} : With \overline{CS} low, the conversion is started on the falling edge of \overline{WR} . Approximately 800 ns (the preset internal time out, t_1) after the \overline{WR} rising edge, the result of the conversion will be strobed into the output latch, provided that \overline{RD} does not occur prior to this time out (see Figures 3, 4). RD Mode RDY : This is an open drain output (no internal pull-up device). RDY will go low after the falling edge of \overline{CS} ; RDY will go TRI-STATE when the result of the conversion is strobed into the output latch. It is used to simplify the interface to a microprocessor system (see Figure 2).
7	Mode	Mode : Mode selection input—it is internally tied to GND through a 50 μ A current source. RD Mode : When mode is low WR-RD Mode : When mode is high
8	\overline{RD}	WR-RD Mode With \overline{CS} low, the TRI-STATE data outputs (DB0-DB7) will be activated when \overline{RD} goes low (see Figure 5). \overline{RD} can also be used to increase the speed of the converter by reading data prior to the preset internal time out (t_1 , ~800 ns). If this is done, the data result transferred to output latch is latched after the falling edge of the \overline{RD} (see Figures 3, 4). RD Mode With \overline{CS} low, the conversion will start with \overline{RD} going low, also \overline{RD} will enable the TRI-STATE data outputs at the completion of the conversion. RDY going TRI-STATE and \overline{INT} going low indicates the completion of the conversion (see Figure 2).

Pin	Name	Function
9	\overline{INT}	WR-RD Mode \overline{INT} going low indicates that the conversion is completed and the data result is in the output latch. \overline{INT} will go low, ~800 ns (the preset internal time out, t_1) after the rising edge of \overline{WR} (see Figure 4); or \overline{INT} will go low after the falling edge of \overline{RD} , if \overline{RD} goes low prior to the 800 ns time out (see Figure 3). \overline{INT} is reset by the rising edge of \overline{RD} or \overline{CS} (see Figures 3, 4). RD Mode \overline{INT} going low indicates that the conversion is completed and the data result is in the output latch. \overline{INT} is reset by the rising edge of \overline{RD} or \overline{CS} (see Figure 2).
10	GND	Ground
11	$V_{REF(-)}$	The bottom of resistor ladder, voltage range: $GND \leq V_{REF(-)} \leq V_{REF(+)}$ (Note 5)
12	$V_{REF(+)}$	The top of resistor ladder, voltage range: $V_{REF(-)} \leq V_{REF(+)} \leq V_{CC}$ (Note 5)
13	\overline{CS}	\overline{CS} must be low in order for the \overline{RD} or \overline{WR} to be recognized by the converter.
14	DB4	TRI-STATE data output—bit 4
15	DB5	TRI-STATE data output—bit 5
16	DB6	TRI-STATE data output—bit 6
17	DB7	TRI-STATE data output—bit 7 (MSB)
18	\overline{OFL}	Overflow output—If the analog input is higher than the $V_{REF(+)}$, \overline{OFL} will be low at the end of conversion. It can be used to cascade 2 or more devices to have more resolution (9, 10-bit). This output is always active and does not go into TRI-STATE as DB0-DB7 do.
19	NC	No connection
20	V_{CC}	Power supply voltage

1.0 Functional Description

1.1 GENERAL OPERATION

The ADC0820 uses two 4-bit flash A/D converters to make an 8-bit measurement (Figure 1). Each flash ADC is made up of 15 comparators which compare the unknown input to a reference ladder to get a 4-bit result. To take a full 8-bit reading, one flash conversion is done to provide the 4 most significant data bits (via the MS flash ADC). Driven by the 4

MSBs, an internal DAC recreates an analog approximation of the input voltage. This analog signal is then subtracted from the input, and the difference voltage is converted by a second 4-bit flash ADC (the LS ADC), providing the 4 least significant bits of the output data word.

The internal DAC is actually a subsection of the MS flash converter. This is accomplished by using the same resistor

1.0 Functional Description (Continued)

ladder for the A/D as well as for generating the DAC signal. The DAC output is actually the tap on the resistor ladder which most closely approximates the analog input. In addition, the "sampled-data" comparators used in the ADC0820 provide the ability to compare the magnitudes of several analog signals simultaneously, without using input summing amplifiers. This is especially useful in the LS flash ADC, where the signal to be converted is an analog difference.

1.2 THE SAMPLED-DATA COMPARATOR

Each comparator in the ADC0820 consists of a CMOS inverter with a capacitively coupled input (Figures 6, 7). Analog switches connect the two comparator inputs to the input capacitor (C) and also connect the inverter's input and output. This device in effect now has one differential input pair. A comparison requires two cycles, one for zeroing the comparator, and another for making the comparison.

In the first cycle, one input switch and the inverter's feedback switch (Figure 6) are closed. In this interval, C is charged to the connected input (V1) less the inverter's bias voltage (V_B, approximately 1.2V). In the second cycle (Figure 7), these two switches are opened and the other (V2) input's switch is closed. The input capacitor now subtracts its stored voltage from the second input and the difference is amplified by the inverter's open loop gain. The inverter's input (V_B') becomes

$$V_B - (V1 - V2) \frac{C}{C + C_S}$$

and the output will go high or low depending on the sign of V_B' - V_B.

The actual circuitry used in the ADC0820 is a simple but important expansion of the basic comparator described above. By adding a second capacitor and another set of switches to the input (Figure 8), the scheme can be expanded to make dual differential comparisons. In this circuit, the feedback switch and one input switch on each capacitor (Z switches) are closed in the zeroing cycle. A comparison is then made

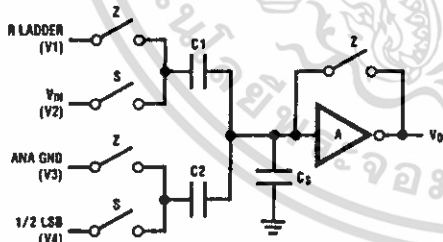


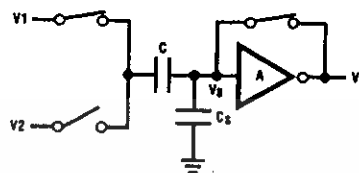
FIGURE 8. ADC0820 Comparator (from MS Flash ADC)

1.3 ARCHITECTURE

In the ADC0820, one bank of 15 comparators is used in each 4-bit flash A/D converter (Figure 12). The MS (most significant) flash ADC also has one additional comparator to detect input overrange. These two sets of comparators operate alternately, with one group in its zeroing cycle while the other is comparing.

When a typical conversion is started, the \overline{WR} line is brought low. At this instant the MS comparators go from zeroing to comparison mode (Figure 11). When \overline{WR} is returned high

by connecting the second input on each capacitor and opening all of the other switches (S switches). The change in voltage at the inverter's input, as a result of the change in charge on each input capacitor, will now depend on both input signal differences.

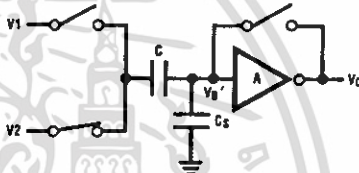


DS006501-12

- V_O = V_B
- V on C = V1 - V_B
- C_S = stray input node capacitor
- V_B = inverter input bias voltage

Zeroing Phase

FIGURE 6. Sampled-Data Comparator



DS006501-13

- V_B' - V_B = (V2 - V1) $\frac{C}{C + C_S}$
- V_O' = $\frac{-A}{C + C_S} [CV2 - CV1]$
- V_O' is dependent on V2 - V1

Compare Phase

FIGURE 7. Sampled-Data Comparator

$$V_O = \frac{-A}{C1 + C2 + C_S} [C1(V2 - V1) + C2(V4 - V3)]$$

$$= \frac{-A}{C1 + C2 + C_S} [\Delta Q_{C1} + \Delta Q_{C2}]$$

DS006501-15

after at least 600 ns, the output from the first set of comparators (the first flash) is decoded and latched. At this point the two 4-bit converters change modes and the LS (least significant) flash ADC enters its compare cycle. No less than 600 ns later, the \overline{RD} line may be pulled low to latch the lower 4 data bits and finish the 8-bit conversion. When \overline{RD} goes low, the flash A/Ds change state once again in preparation for the next conversion.

Figure 11 also outlines how the converter's interface timing relates to its analog input (V_{IN}). In \overline{WR} - \overline{RD} mode, V_{IN} is mea-

1.0 Functional Description (Continued)

sure while \overline{WR} is low. In RD mode, sampling occurs during the first 800 ns of \overline{RD} . Because of the input connections to the ADC0820's LS and MS comparators, the converter has the ability to sample V_{IN} at one instant (Section 2.4), despite the fact that two separate 4-bit conversions are being done. More specifically, when \overline{WR} is low the MS flash is in compare mode (connected to V_{IN}), and the LS flash is in zero mode (also connected to V_{IN}). Therefore both flash ADCs sample V_{IN} at the same time.

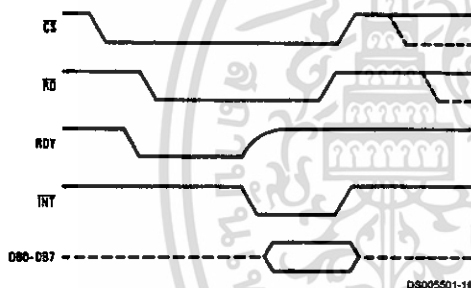
1.4 DIGITAL INTERFACE

The ADC0820 has two basic interface modes which are selected by strapping the MODE pin high or low.

RD Mode

With the MODE pin grounded, the converter is set to Read mode. In this configuration, a complete conversion is done by pulling \overline{RD} low until output data appears. An \overline{INT} line is provided which goes low at the end of the conversion as well as a RDY output which can be used to signal a processor that the converter is busy or can also serve as a system Transfer Acknowledge signal.

RD Mode (Pin 7 is Low)



When in RD mode, the comparator phases are internally triggered. At the falling edge of \overline{RD} , the MS flash converter goes from zero to compare mode and the LS ADC's comparators enter their zero cycle. After 800 ns, data from the MS flash is latched and the LS flash ADC enters compare mode. Following another 800 ns, the lower 4 bits are recovered.

WR then RD Mode

With the MODE pin tied high, the A/D will be set up for the WR-RD mode. Here, a conversion is started with the \overline{WR} input; however, there are two options for reading the output data which relate to interface timing. If an interrupt driven scheme is desired, the user can wait for \overline{INT} to go low before reading the conversion result (Figure 10). \overline{INT} will typically go low 800 ns after \overline{WR} 's rising edge. However, if a shorter

conversion time is desired, the processor need not wait for \overline{INT} and can exercise a read after only 600 ns (Figure 9). If this is done, \overline{INT} will immediately go low and data will appear at the outputs.

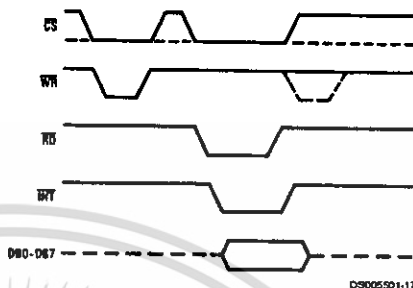


FIGURE 9. WR-RD Mode (Pin 7 is High and $t_{RD} < t_t$)

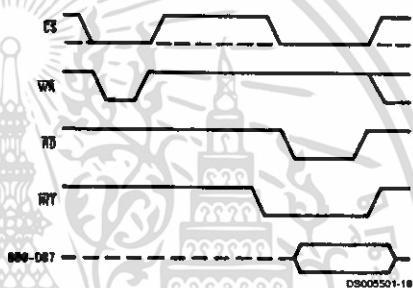
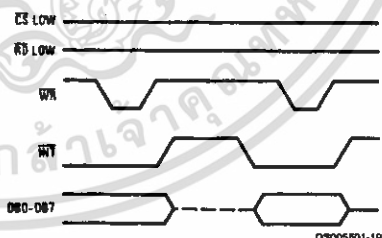


FIGURE 10. WR-RD Mode (Pin 7 is High and $t_{RD} > t_t$)

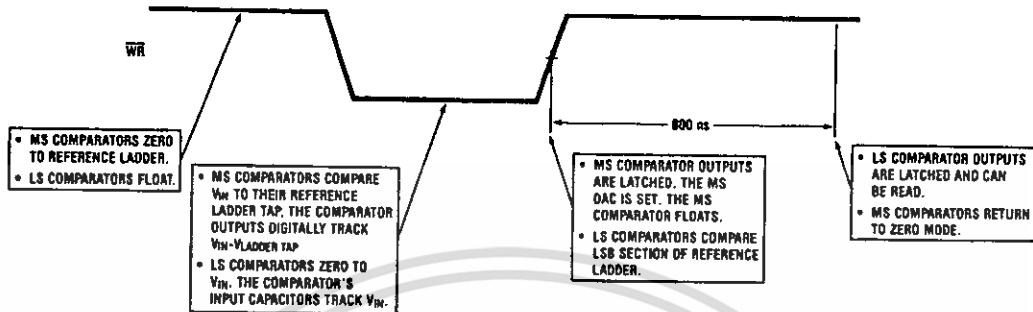
Stand-Alone

For stand-alone operation in WR-RD mode, \overline{CS} and \overline{RD} can be tied low and a conversion can be started with \overline{WR} . Data will be valid approximately 800 ns following \overline{WR} 's rising edge.

WR-RD Mode (Pin 7 is High) Stand-Alone Operation



1.0 Functional Description (Continued)



Note: MS means most significant
LS means least significant

DS005501-20

FIGURE 11. Operating Sequence (WR-RD Mode)

OTHER INTERFACE CONSIDERATIONS

In order to maintain conversion accuracy, \overline{WR} has a maximum width spec of 50 μ s. When the MS flash ADC's sampled-data comparators (Section 1.2) are in comparison mode (\overline{WR} is low), the input capacitors (C, Figure 8) must hold their charge. Switch leakage and inverter bias current can cause errors if the comparator is left in this phase for too long.

Since the MS flash ADC enters its zeroing phase at the end of a conversion (Section 1.3), a new conversion cannot be started until this phase is complete. The minimum spec for this time (t_p , Figures 2, 3, 4, 5) is 500 ns.

Detailed Block Diagram

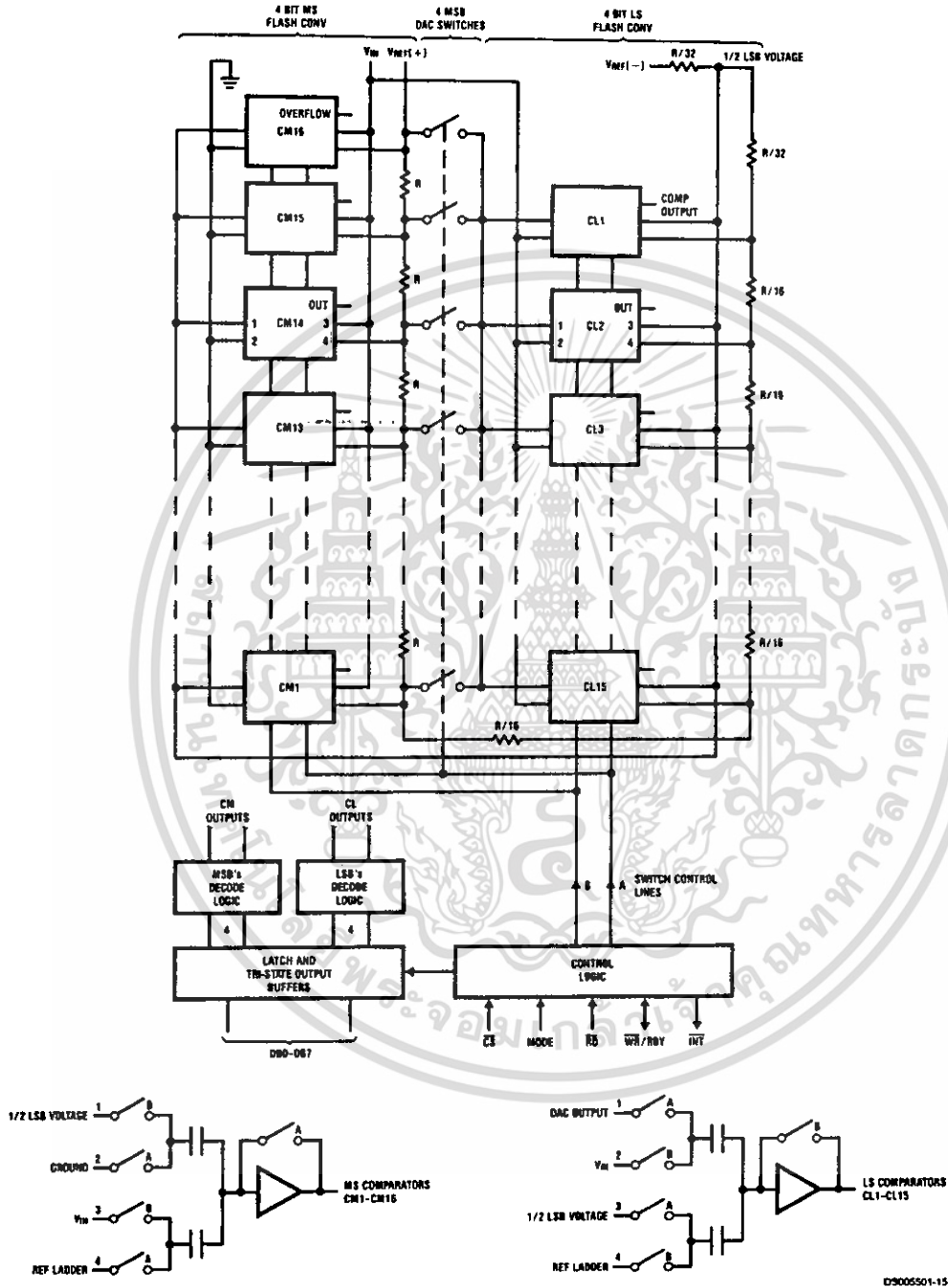


FIGURE 12.

D9005501-15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.0 Analog Considerations

2.1 REFERENCE AND INPUT

The two V_{REF} inputs of the ADC0820 are fully differential and define the zero to full-scale input range of the A/D converter. This allows the designer to easily vary the span of the analog input since this range will be equivalent to the voltage difference between $V_{IN}(+)$ and $V_{IN}(-)$. By reducing $V_{REF}(V_{REF}=V_{REF(+)}-V_{REF}(-))$ to less than 5V, the sensitivity of the converter can be increased (i.e., if $V_{REF}=2V$ then 1 LSB=7.8 mV). The input/reference arrangement also facilitates ratiometric operation and in many cases the chip power supply can be used for transducer power as well as the V_{REF} source.

This reference flexibility lets the input span not only be varied but also offset from zero. The voltage at $V_{REF}(-)$ sets the input level which produces a digital output of all zeroes. Though V_{IN} is not itself differential, the reference design affords nearly differential-input capability for most measurement applications. Figure 13 shows some of the configurations that are possible.

2.2 INPUT CURRENT

Due to the unique conversion techniques employed by the ADC0820, the analog input behaves somewhat differently than in conventional devices. The A/D's sampled-data comparators take varying amounts of input current depending on which cycle the conversion is in.

The equivalent input circuit of the ADC0820 is shown in Figure 14. When a conversion starts (\overline{WR} low, WR-RD mode), all input switches close, connecting V_{IN} to thirty-one 1 pF capacitors. Although the two 4-bit flash circuits are not both in their compare cycle at the same time, V_{IN} still sees all input capacitors at once. This is because the MS flash converter is connected to the input during its compare interval and the LS flash is connected to the input during its zeroing phase (Section 1.3). In other words, the LS ADC uses V_{IN} as its zero-phase input.

The input capacitors must charge to the input voltage through the on resistance of the analog switches (about 5 k Ω to 10 k Ω). In addition, about 12 pF of input stray capacitance must also be charged. For large source resistances, the analog input can be modeled as an RC network as shown in Figure 15. As R_S increases, it will take longer for the input capacitance to charge.

In RD mode, the input switches are closed for approximately 800 ns at the start of the conversion. In WR-RD mode, the time that the switches are closed to allow this charging is the time that \overline{WR} is low. Since other factors force this time to be at least 600 ns, input time constants of 100 ns can be accommodated without special consideration. Typical total input capacitance values of 45 pF allow R_S to be 1.5 k Ω without lengthening \overline{WR} to give V_{IN} more time to settle.

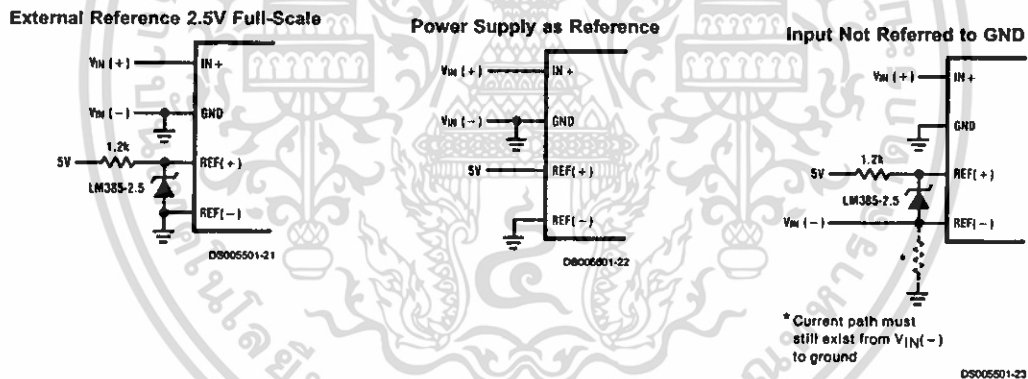


FIGURE 13. Analog Input Options

2.0 Analog Considerations (Continued)

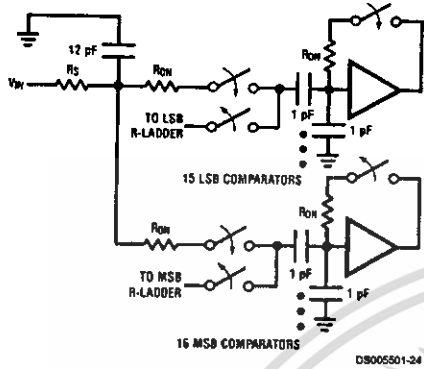


FIGURE 14.

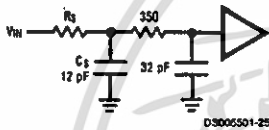


FIGURE 15.

2.3 INPUT FILTERING

It should be made clear that transients in the analog input signal, caused by charging current flowing into V_{IN} , will not degrade the A/D's performance in most cases. In effect the ADC0820 does not "look" at the input when these transients occur. The comparators' outputs are not latched while WR is low, so at least 600 ns will be provided to charge the ADC's input capacitance. It is therefore not necessary to filter out these transients by putting an external cap on the V_{IN} terminal.

2.4 INHERENT SAMPLE-HOLD

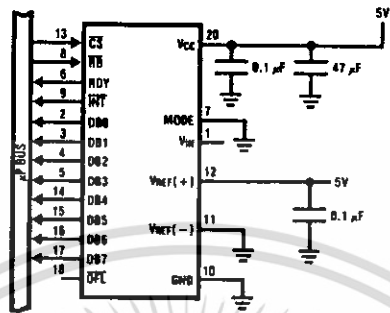
Another benefit of the ADC0820's input mechanism is its ability to measure a variety of high speed signals without the help of an external sample-and-hold. In a conventional SAR type converter, regardless of its speed, the input must remain at least $\frac{1}{2}$ LSB stable throughout the conversion process if full accuracy is to be maintained. Consequently, for many high speed signals, this signal must be externally sampled, and held stationary during the conversion.

Sampled-data comparators, by nature of their input switching, already accomplish this function to a large degree (Section 1.2). Although the conversion time for the ADC0820 is 1.5 μ s, the time through which V_{IN} must be $\frac{1}{2}$ LSB stable is much smaller. Since the MS flash ADC uses V_{IN} as its "compare" input and the LS ADC uses V_{IN} as its "zero" input, the ADC0820 only "samples" V_{IN} when WR is low (Sections 1.3 and 2.2). Even though the two flashes are not done simultaneously, the analog signal is measured at one instant. The value of V_{IN} approximately 100 ns after the rising edge of WR (100 ns due to internal logic prop delay) will be the measured value.

Input signals with slew rates typically below 100 mV/ μ s can be converted without error. However, because of the input time constants, and charge injection through the opened comparator input switches, faster signals may cause errors. Still, the ADC0820's loss in accuracy for a given increase in signal slope is far less than what would be witnessed in a conventional successive approximation device. An SAR type converter with a conversion time as fast as 1 μ s would still not be able to measure a 5V 1 kHz sine wave without the aid of an external sample-and-hold. The ADC0820, with no such help, can typically measure 5V, 7 kHz waveforms.

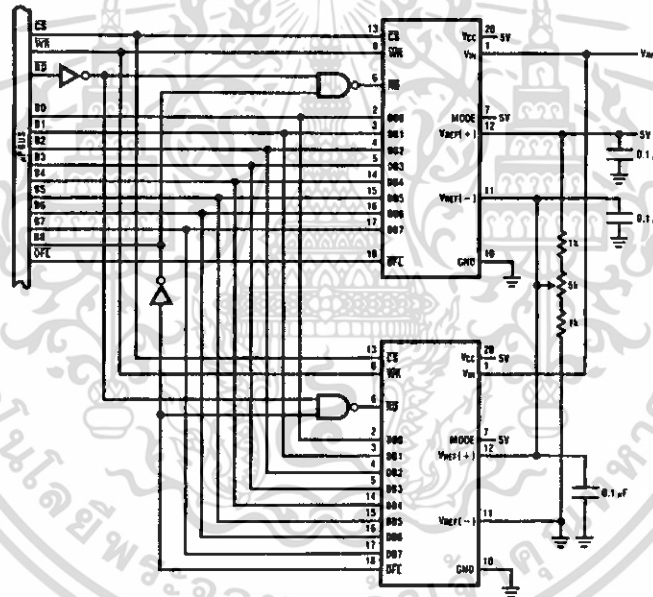
3.0 Typical Applications

8-Bit Resolution Configuration



DS005501-26

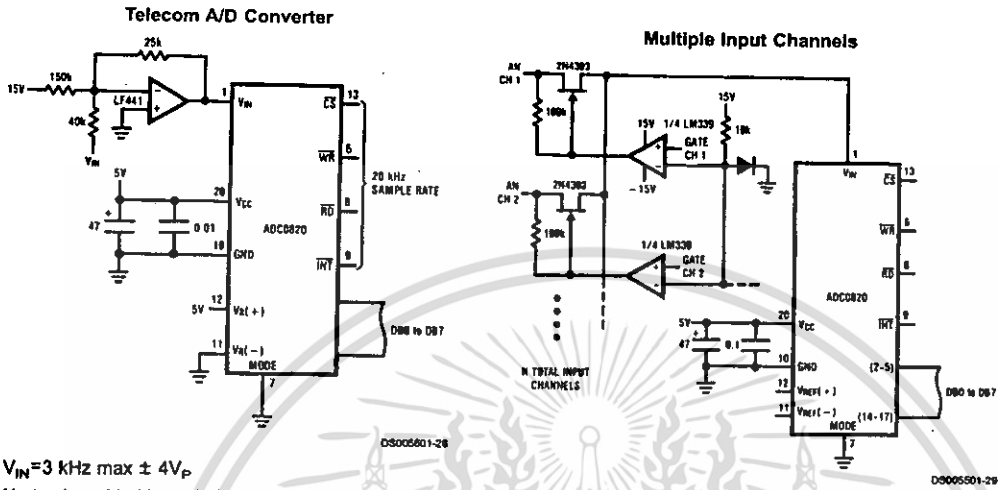
9-Bit Resolution Configuration



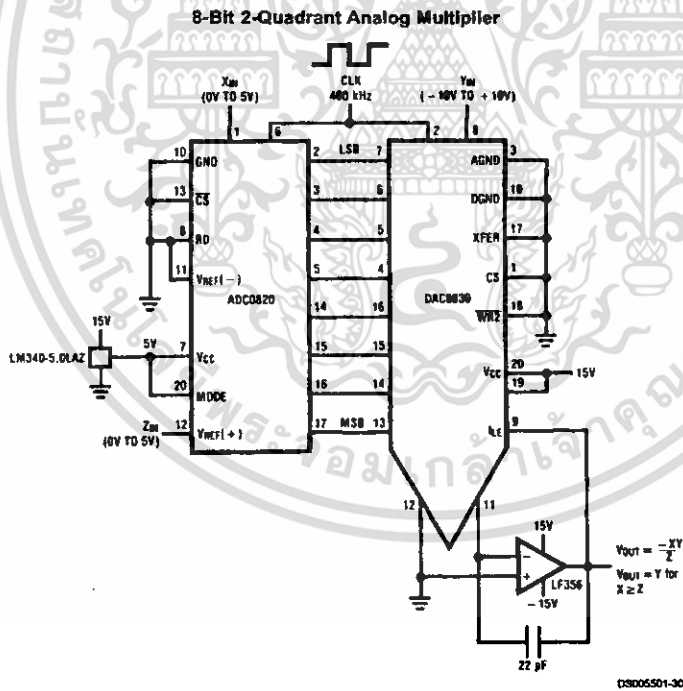
DS005501-27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.0 Typical Applications (Continued)

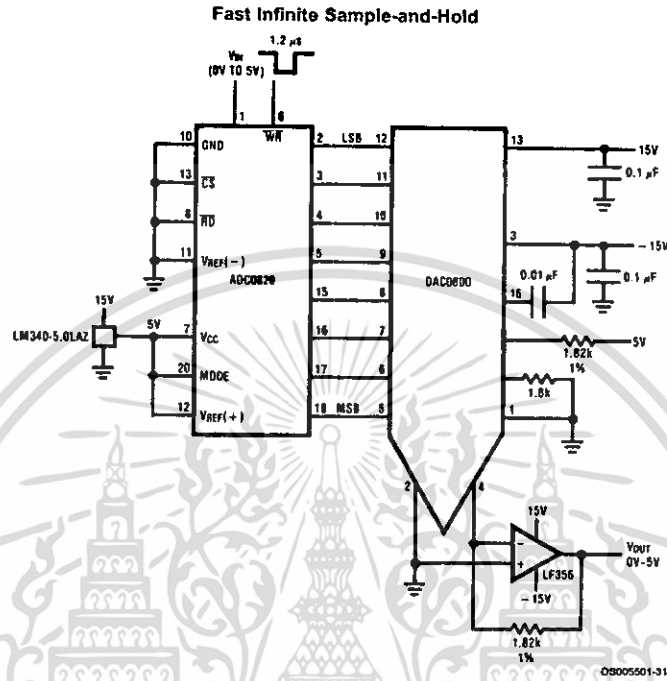


- $V_{IN} = 3 \text{ kHz max } \pm 4V_P$
- No track-and-hold needed
- Low power consumption



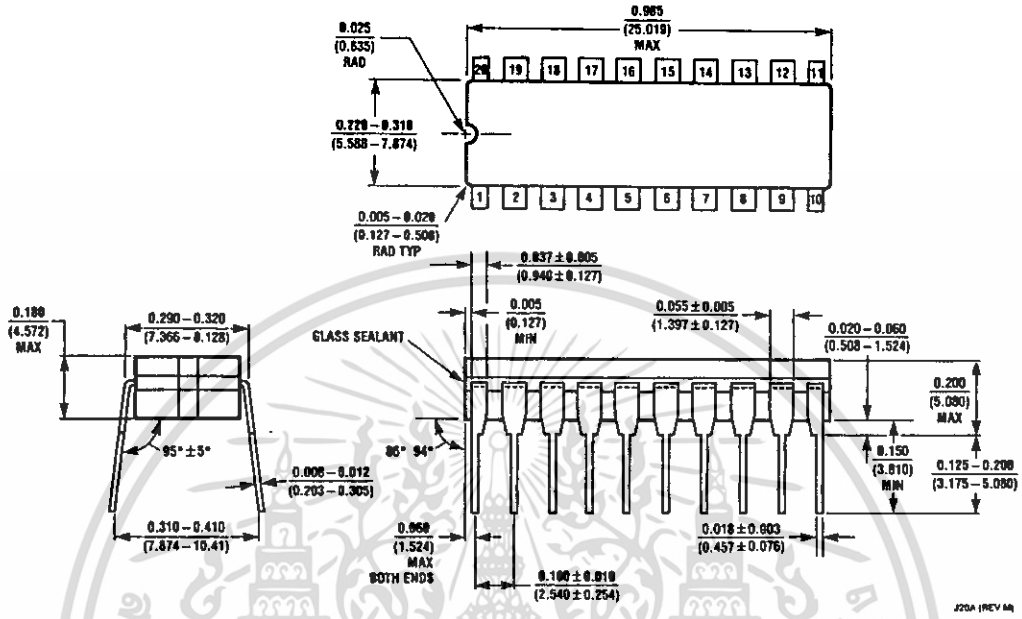
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.0 Typical Applications (Continued)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) unless otherwise noted

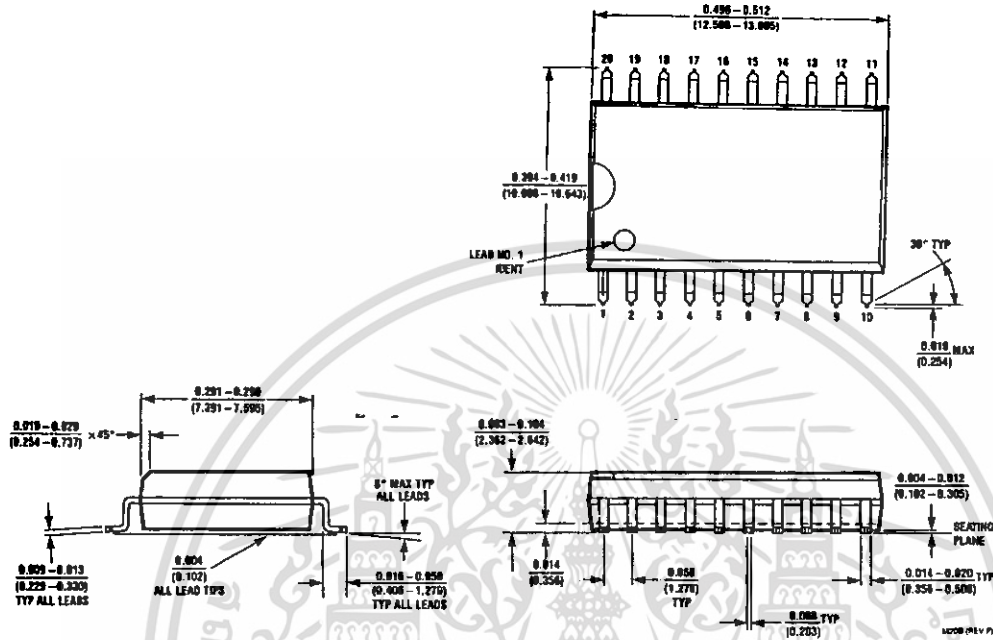


Hermetic Dual-In-Line Package (J)
 Order Number ADC0820CCJ
 NS Package Number J20A

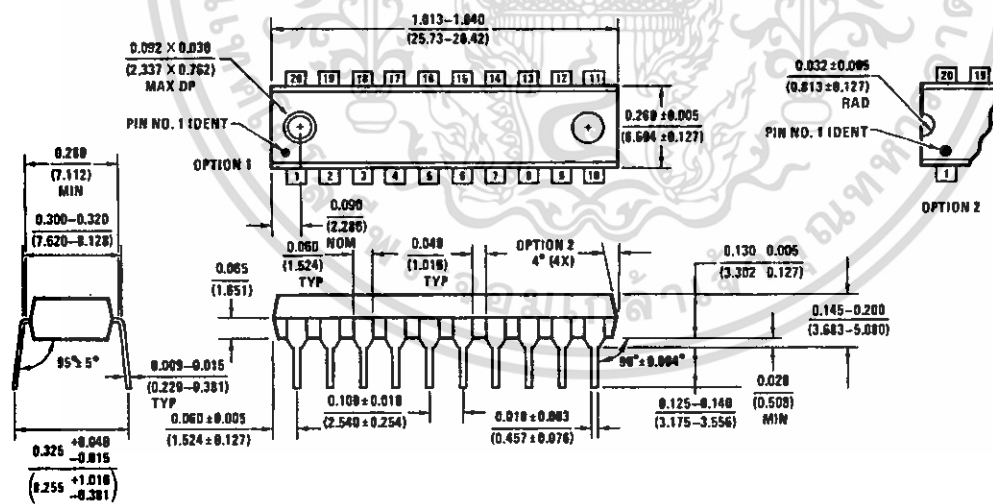
J20A (REV 04)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) unless otherwise noted (Continued)



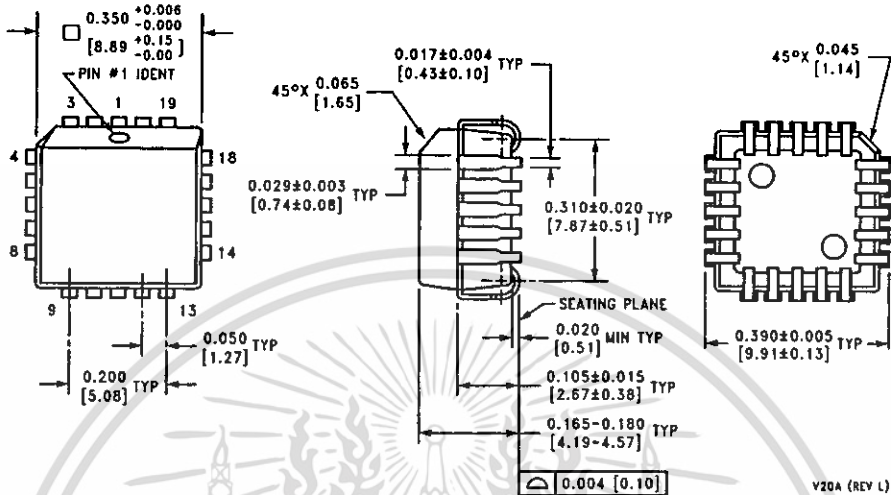
SO Package (M)
 Order Number ADC0820BCWM, ADC0820CCWM or ADC0820CIWM
 NS Package Number M20B



Molded Dual-In-Line Package (N)
 Order Number ADC0820BCN or ADC0820CCN
 NS Package Number N20A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) unless otherwise noted (Continued)



Molded Chip Carrier Package (V)
 Order Number ADC0820BCV
 NS Package Number V20A

V20A (REV L)

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation Americas
 Tel: 1-800-272-4959
 Fax: 1-800-737-7018
 Email: support@nsc.com

National Semiconductor Europe
 Fax: +49 (0) 1 80-530 85 86
 Email: europe.support@nsc.com
 Deutsch Tel: +49 (0) 1 80-530 85 85
 English Tel: +49 (0) 1 80-532 78 32
 Français Tel: +49 (0) 1 80-532 83 58
 Italiano Tel: +49 (0) 1 80-534 16 80

National Semiconductor Asia Pacific Customer Response Group
 Tel: 65-2544468
 Fax: 65-2504468
 Email: sen.support@nsc.com

National Semiconductor Japan Ltd.
 Tel: 81-3-5639-7560
 Fax: 81-3-5639-7507

www.national.com

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DAC0808

8-Bit D/A Converter

General Description

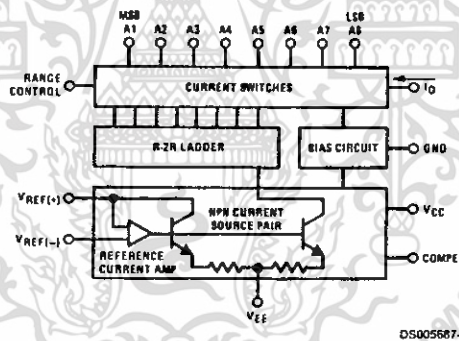
The DAC0808 is an 8-bit monolithic digital-to-analog converter (DAC) featuring a full scale output current settling time of 150 ns while dissipating only 33 mW with $\pm 5V$ supplies. No reference current (I_{REF}) trimming is required for most applications since the full scale output current is typically ± 1 LSB of $255 I_{REF}/256$. Relative accuracies of better than $\pm 0.19\%$ assure 8-bit monotonicity and linearity while zero level output current of less than $4 \mu A$ provides 8-bit zero accuracy for $I_{REF} \geq 2 \text{ mA}$. The power supply currents of the DAC0808 is independent of bit codes, and exhibits essentially constant device characteristics over the entire supply voltage range.

The DAC0808 will interface directly with popular TTL, DTL or CMOS logic levels, and is a direct replacement for the MC1508/MC1408. For higher speed applications, see DAC0800 data sheet.

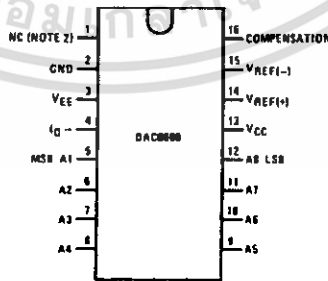
Features

- Relative accuracy: $\pm 0.19\%$ error maximum
- Full scale current match: ± 1 LSB typ
- Fast settling time: 150 ns typ
- Noninverting digital inputs are TTL and CMOS compatible
- High speed multiplying input slew rate: $8 \text{ mA}/\mu\text{s}$
- Power supply voltage range: $\pm 4.5V$ to $\pm 18V$
- Low power consumption: $33 \text{ mW} @ \pm 5V$

Block and Connection Diagrams



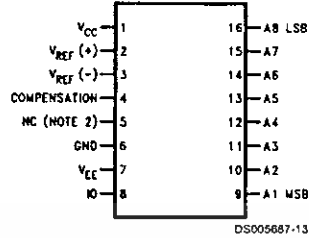
Dual-In-Line Package



Top View
Order Number DAC0808
 See NS Package M16A or N16A

Block and Connection Diagrams (Continued)

Small-Outline Package



Ordering Information

ACCURACY	OPERATING TEMPERATURE RANGE	N PACKAGE (N16A) (Note 1)		SO PACKAGE (M16A)
		DAC0808LCN	MC1408P8	DAC0808LCM
8-bit	0°C ≤ T _A ≤ +75°C			

Note 1: Devices may be ordered by using either order number.



Absolute Maximum Ratings (Note 2)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Power Supply Voltage

V_{CC}	+18 V_{DC}
V_{EE}	-18 V_{DC}
Digital Input Voltage, V5-V12	-10 V_{DC} to +18 V_{DC}
Applied Output Voltage, V_O	-11 V_{DC} to +18 V_{DC}
Reference Current, I_{14}	5 mA
Reference Amplifier Inputs, V14, V15	V_{CC}, V_{EE}
Power Dissipation (Note 4)	1000 mW
ESD Susceptibility (Note 5)	TBD

Storage Temperature Range

-65°C to +150°C

Lead Temp. (Soldering, 10 seconds)

Dual-In-Line Package (Plastic)	260°C
Dual-In-Line Package (Ceramic)	300°C
Surface Mount Package	
Vapor Phase (60 seconds)	215°C
Infrared (15 seconds)	220°C

Operating Ratings

Temperature Range
DAC0808

$T_{MIN} \leq T_A \leq T_{MAX}$
 $0 \leq T_A \leq +75^\circ C$

Electrical Characteristics

($V_{CC} = 5V, V_{EE} = -15 V_{DC}, V_{REF}/R14 = 2 mA$, and all digital inputs at high logic level unless otherwise noted.)

Symbol	Parameter	Conditions	Min	Typ	Max	Units
E_r	Relative Accuracy (Error Relative to Full Scale I_O)	(Figure 4)				%
	DAC0808LC (LM1408-8)				± 0.19	%
	Settling Time to Within 1/2 LSB (Includes t_{PLH})	$T_A = 25^\circ C$ (Note 7), (Figure 5)		150		ns
t_{PLH}, t_{PHL}	Propagation Delay Time	$T_A = 25^\circ C$, (Figure 5)		30	100	ns
TCI_O	Output Full Scale Current Drift			± 20		ppm/°C
MSB V_{IH} V_{IL}	Digital Input Logic Levels	(Figure 3)				
	High Level, Logic "1" Low Level, Logic "0"		2		0.8	V_{DC} V_{DC}
MSB	Digital Input Current	(Figure 3)				
	High Level Low Level	$V_{IH} = 5V$ $V_{IL} = 0.8V$		0 -0.003	0.040 -0.8	mA mA
I_{15}	Reference Input Bias Current	(Figure 3)		-1	-3	μA
	Output Current Range	(Figure 3)				
		$V_{EE} = -5V$ $V_{EE} = -15V, T_A = 25^\circ C$	0 0	2.0 2.0	2.1 4.2	mA mA
I_O	Output Current	$V_{REF} = 2.000V$, $R14 = 1000\Omega$, (Figure 3)	1.9	1.99	2.1	mA
	Output Current, All Bits Low	(Figure 3)		0	4	μA
	Output Voltage Compliance (Note 3)	$E_r \leq 0.19\%$, $T_A = 25^\circ C$			-0.55, +0.4 -5.0, +0.4	V_{DC} V_{DC}
SRI_{REF}	Reference Current Slew Rate	(Figure 6)	4	8		mA/ μs
	Output Current Power Supply Sensitivity	$-5V \leq V_{EE} \leq -16.5V$		0.05	2.7	$\mu A/V$
I_{CC} I_{EE}	Power Supply Current (All Bits Low)	(Figure 3)				
				2.3 -4.3	22 -13	mA mA
V_{CC} V_{EE}	Power Supply Voltage Range	$T_A = 25^\circ C$, (Figure 3)	4.5 -4.5	5.0 -15	5.5 -16.5	V_{DC} V_{DC}
		Power Dissipation				

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Characteristics (Continued)

($V_{CC} = 5V$, $V_{EE} = -15V_{DC}$, $V_{REF}/R_{14} = 2\text{ mA}$, and all digital inputs at high logic level unless otherwise noted.)

Symbol	Parameter	Conditions	Min	Typ	Max	Units
	All Bits Low	$V_{CC} \approx 5V$, $V_{EE} = -5V$		33	170	mW
		$V_{CC} \approx 5V$, $V_{EE} = -15V$		106	305	mW
	All Bits High	$V_{CC} \approx 15V$, $V_{EE} = -5V$		90		mW
		$V_{CC} \approx 15V$, $V_{EE} = -15V$		160		mW

Note 2: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its specified operating conditions.

Note 3: Range control is not required.

Note 4: The maximum power dissipation must be derated at elevated temperatures and is dictated by $T_{JMAX} - 0_{JA}$, and the ambient temperature, T_A . The maximum allowable power dissipation at any temperature is $P_D = (T_{JMAX} - T_A)/\theta_{JA}$ or the number given in the Absolute Maximum Ratings, whichever is lower. For this device, $T_{JMAX} = 125^\circ\text{C}$, and the typical junction-to-ambient thermal resistance of the dual-in-line J package when the board mounted is 100°C/W . For the dual-in-line N package, this number increases to 175°C/W and for the small outline M package this number is 100°C/W .

Note 5: Human body model, 100 pF discharged through a 1.5 kΩ resistor.

Note 6: All current switches are tested to guarantee at least 50% of rated current.

Note 7: All bits switched.

Note 8: Pin-out numbers for the DAL080X represent the dual-in-line package. The small outline package pinout differs from the dual-in-line package.

Typical Application

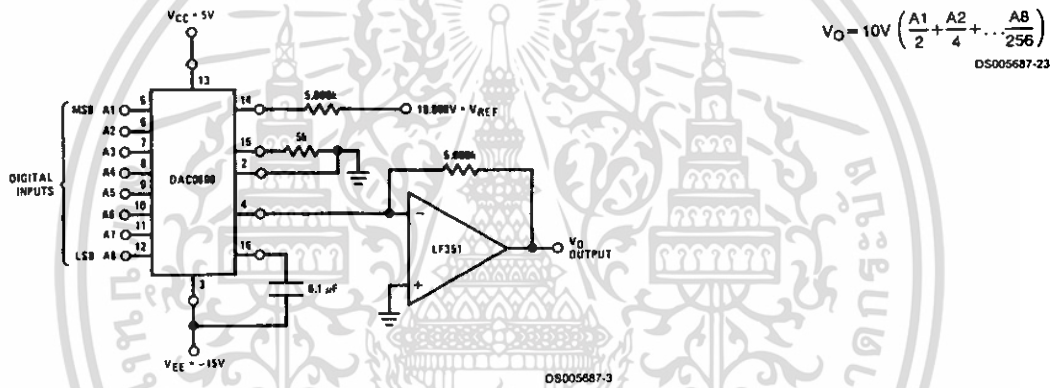
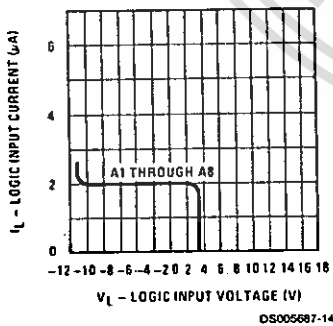


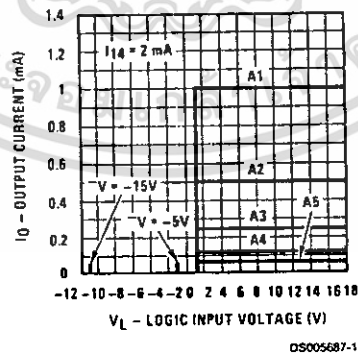
FIGURE 1. +10V Output Digital to Analog Converter (Note 8)

Typical Performance Characteristics $V_{CC} = 5V$, $V_{EE} = -15V$, $T_A = 25^\circ\text{C}$, unless otherwise noted

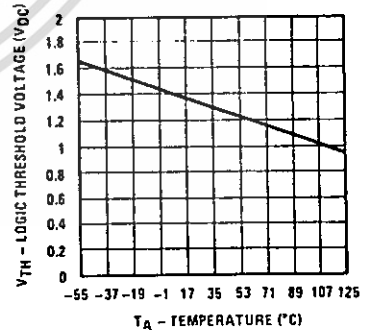
Logic Input Current vs Input Voltage



Bit Transfer Characteristics

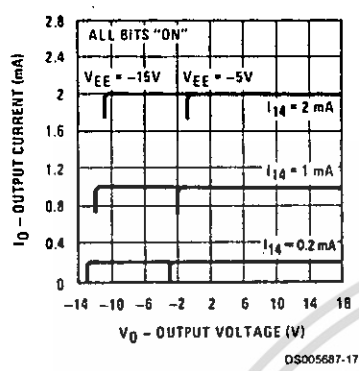


Logic Threshold Voltage vs Temperature



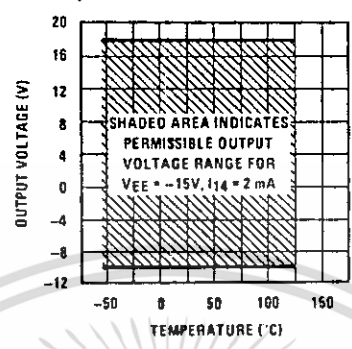
Typical Performance Characteristics $V_{CC} = 5V$, $V_{EE} = -15V$, $T_A = 25^\circ C$, unless otherwise noted (Continued)

Output Current vs Output Voltage (Output Voltage Compliance)



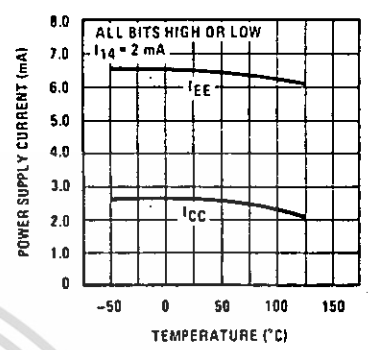
DS005687-17

Output Voltage Compliance vs Temperature



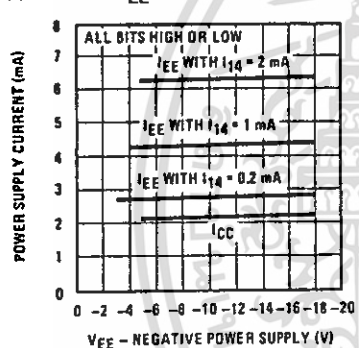
DS005687-18

Typical Power Supply Current vs Temperature



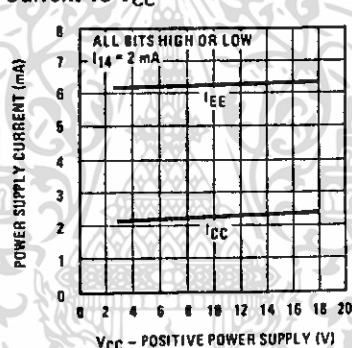
DS005687-19

Typical Power Supply Current vs VEE



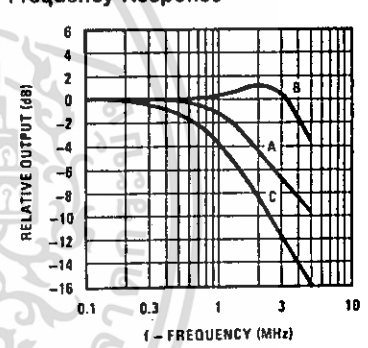
DS005687-20

Typical Power Supply Current vs VCC



DS005687-21

Reference Input Frequency Response



DS005687-22

Unless otherwise specified: $R_{14} = R_{15} = 1\text{ k}\Omega$, $C = 15\text{ pF}$, pin 16 to V_{EE} ; $R_L = 50\Omega$, pin 4 to ground.
Curve A: Large Signal Bandwidth Method of Figure 7, $V_{REF} = 2\text{ Vp-p}$ offset 1V above ground.
Curve B: Small Signal Bandwidth Method of Figure 7, $R_L = 250\Omega$, $V_{REF} = 50\text{ mVp-p}$ offset 200 mV above ground.
Curve C: Large and Small Signal Bandwidth Method of Figure 9 (no op amp, $R_L = 50\Omega$), $R_S = 50\Omega$, $V_{REF} = 2V$, $V_S = 100\text{ mVp-p}$ centered at 0V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

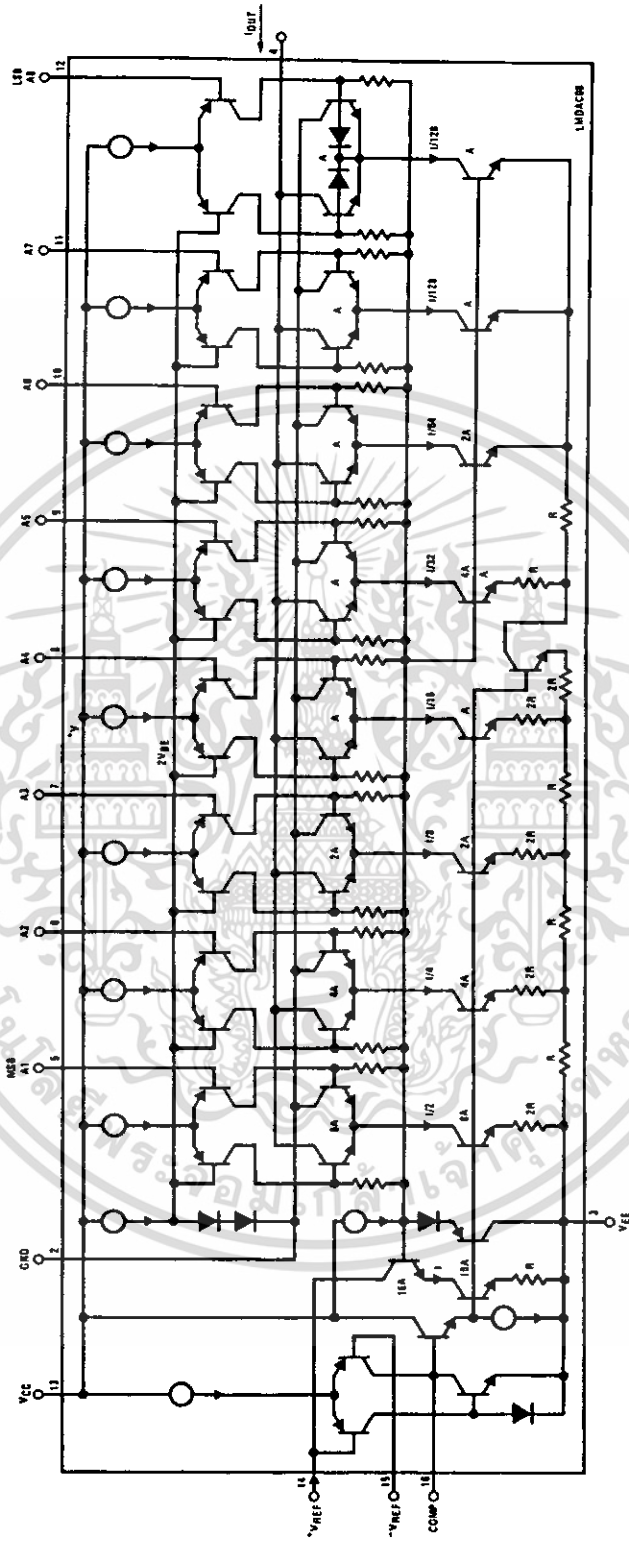
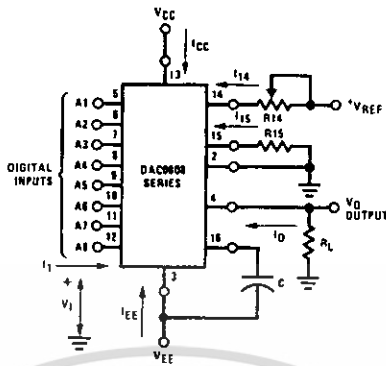


FIGURE 2. Equivalent Circuit of the DAC0808 Series (Note 8)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Test Circuits



DS005687-6

V_I and I_1 apply to inputs A1–A8.

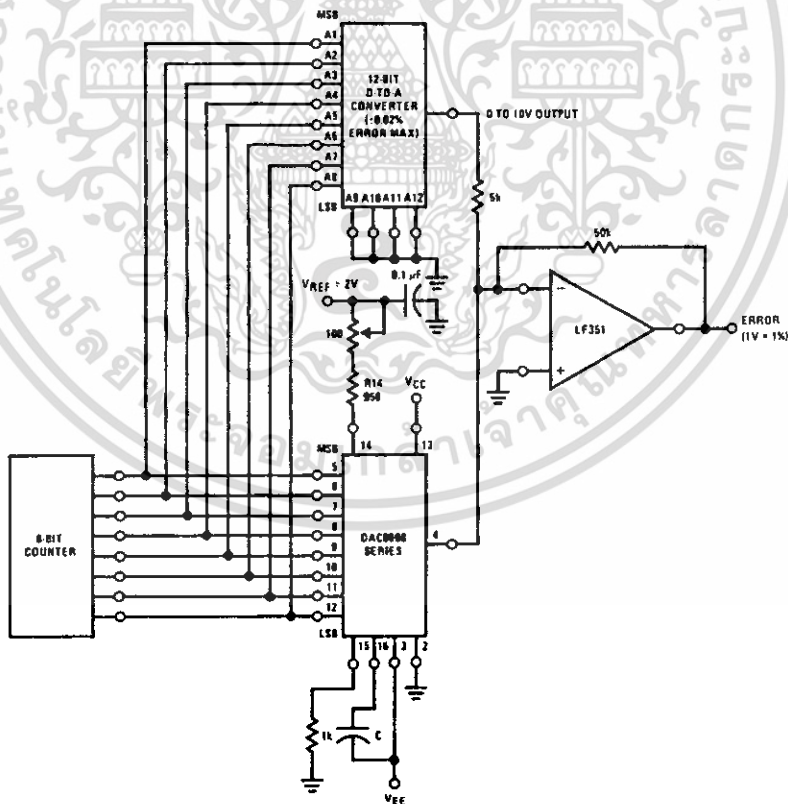
The resistor tied to pin 15 is to temperature compensate the bias current and may not be necessary for all applications.

$$I_O = K \left(\frac{A_1}{2} + \frac{A_2}{4} + \frac{A_3}{8} + \frac{A_4}{16} + \frac{A_5}{32} + \frac{A_6}{64} + \frac{A_7}{128} + \frac{A_8}{256} \right)$$

where $K \approx \frac{V_{REF}}{R_{14}}$

and $A_N = "1"$ if A_N is at high level
 $A_N = "0"$ if A_N is at low level

FIGURE 3. Notation Definitions Test Circuit (Note 8)



DS005687-7

FIGURE 4. Relative Accuracy Test Circuit (Note 8)

Test Circuits (Continued)

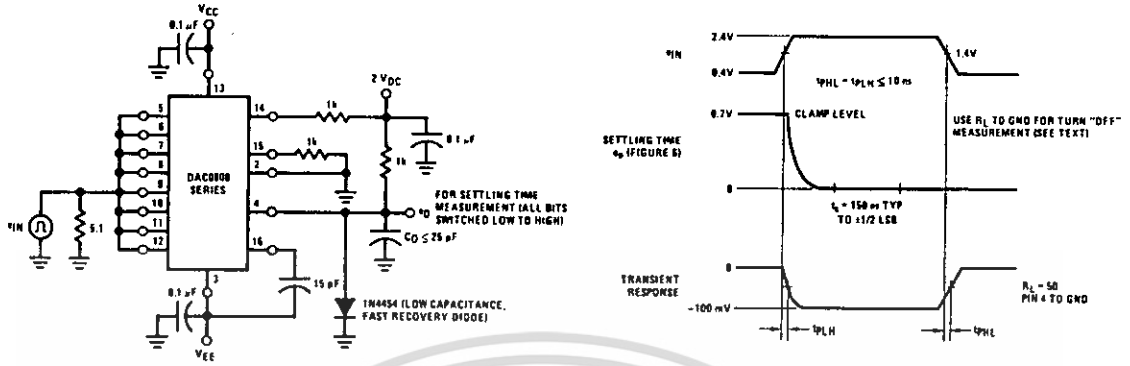


FIGURE 5. Transient Response and Settling Time (Note 8)

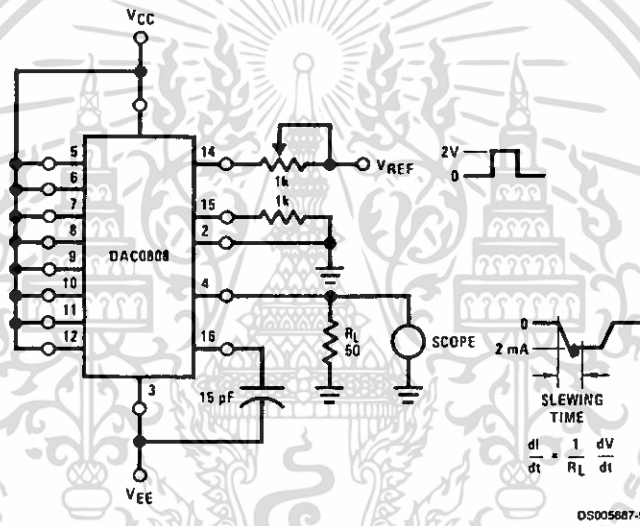


FIGURE 6. Reference Current Slew Rate Measurement (Note 8)

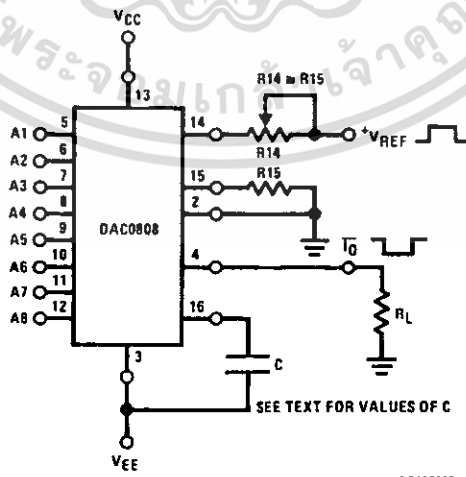


FIGURE 7. Positive V_{REF} (Note 8)

Test Circuits (Continued)

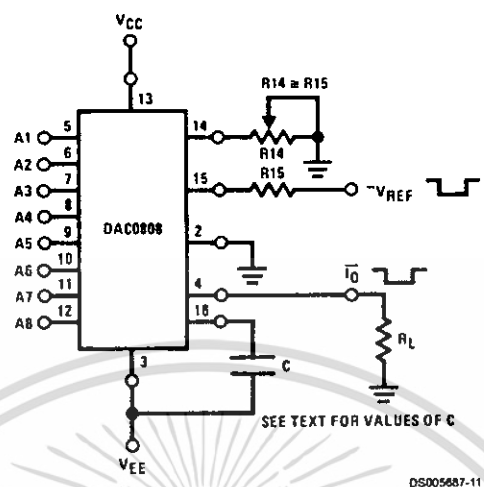


FIGURE 8. Negative V_{REF} (Note 8)

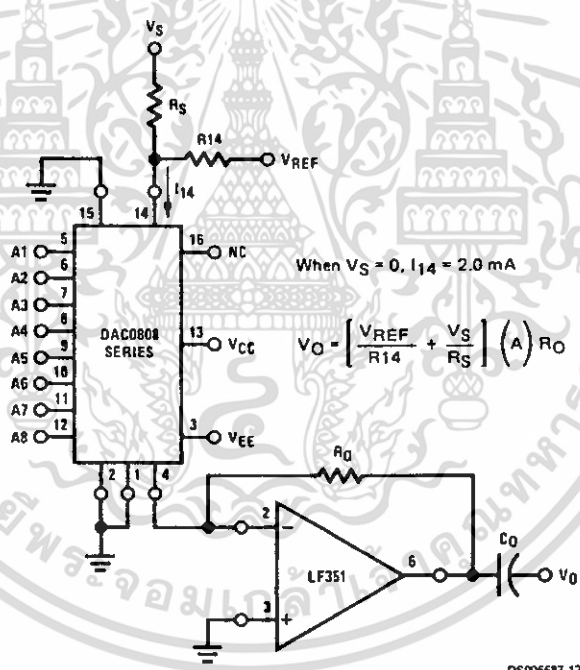


FIGURE 9. Programmable Gain Amplifier or Digital Attenuator Circuit (Note 8)

Application Hints

REFERENCE AMPLIFIER DRIVE AND COMPENSATION

The reference amplifier provides a voltage at pin 14 for converting the reference voltage to a current, and a turn-around circuit or current mirror for feeding the ladder. The reference amplifier input current, I₁₄, must always flow into pin 14, regardless of the set-up method or reference voltage polarity.

Connections for a positive voltage are shown in Figure 7. The reference voltage source supplies the full current I₁₄.

For bipolar reference signals, as in the multiplying mode, R₁₅ can be tied to a negative voltage corresponding to the minimum input level. It is possible to eliminate R₁₅ with only a small sacrifice in accuracy and temperature drift.

The compensation capacitor value must be increased with increases in R₁₄ to maintain proper phase margin; for R₁₄ values of 1, 2.5 and 5 kΩ, minimum capacitor values are 15, 37 and 75 pF. The capacitor may be tied to either V_{EE} or ground, but using V_{EE} increases negative supply rejection.

A negative reference voltage may be used if R₁₄ is grounded and the reference voltage is applied to R₁₅ as shown in Figure 8. A high input impedance is the main

Application Hints (Continued)

advantage of this method. Compensation involves a capacitor to V_{EE} on pin 16, using the values of the previous paragraph. The negative reference voltage must be at least 4V above the V_{EE} supply. Bipolar input signals may be handled by connecting R14 to a positive reference voltage equal to the peak positive input level at pin 15.

When a DC reference voltage is used, capacitive bypass to ground is recommended. The 5V logic supply is not recommended as a reference voltage. If a well regulated 5V supply which drives logic is to be used as the reference, R14 should be decoupled by connecting it to 5V through another resistor and bypassing the junction of the 2 resistors with 0.1 μ F to ground. For reference voltages greater than 5V, a clamp diode is recommended between pin 14 and ground.

If pin 14 is driven by a high impedance such as a transistor current source, none of the above compensation methods apply and the amplifier must be heavily compensated, decreasing the overall bandwidth.

OUTPUT VOLTAGE RANGE

The voltage on pin 4 is restricted to a range of -0.55 to 0.4V when $V_{EE} = -5V$ due to the current switching methods employed in the DAC0808.

The negative output voltage compliance of the DAC0808 is extended to -5V where the negative supply voltage is more negative than -10V. Using a full-scale current of 1.992 mA and load resistor of 2.5 k Ω between pin 4 and ground will yield a voltage output of 256 levels between 0 and -4.980V. Floating pin 1 does not affect the converter speed or power dissipation. However, the value of the load resistor determines the switching time due to increased voltage swing. Values of R_L up to 500 Ω do not significantly affect performance, but a 2.5 k Ω load increases worst-case settling time to 1.2 μ s (when all bits are switched ON). Refer to the subsequent text section on Settling Time for more details on output loading.

OUTPUT CURRENT RANGE

The output current maximum rating of 4.2 mA may be used only for negative supply voltages more negative than -8V, due to the increased voltage drop across the resistors in the reference current amplifier.

ACCURACY

Absolute accuracy is the measure of each output current level with respect to its intended value, and is dependent upon relative accuracy and full-scale current drift. Relative accuracy is the measure of each output current level as a fraction of the full-scale current. The relative accuracy of the DAC0808 is essentially constant with temperature due to the excellent temperature tracking of the monolithic resistor ladder.

The reference current may drift with temperature, causing a change in the absolute accuracy of output current. However, the DAC0808 has a very low full-scale current drift with temperature.

The DAC0808 series is guaranteed accurate to within $\pm 1/2$ LSB at a full-scale output current of 1.992 mA. This corresponds to a reference amplifier output current drive to the ladder network of 2 mA, with the loss of 1 LSB (8 μ A) which is the ladder remainder shunted to ground. The input current to pin 14 has a guaranteed value of between 1.9 and 2.1 mA, allowing some mismatch in the NPN current source pair. The accuracy test circuit is shown in Figure 4. The 12-bit converter is calibrated for a full-scale output current of 1.992 mA. This is an optional step since the DAC0808 accuracy is essentially the same between 1.5 and 2.5 mA. Then the DAC0808 circuits' full-scale current is trimmed to the same value with R14 so that a zero value appears at the error amplifier output. The counter is activated and the error band may be displayed on an oscilloscope, detected by comparators, or stored in a peak detector.

Two 8-bit D-to-A converters may not be used to construct a 16-bit accuracy D-to-A converter. 16-bit accuracy implies a total error of $\pm 1/2$ of one part in 65,536 or $\pm 0.00076\%$, which is much more accurate than the $\pm 0.019\%$ specification provided by the DAC0808.

MULTIPLYING ACCURACY

The DAC0808 may be used in the multiplying mode with 8-bit accuracy when the reference current is varied over a range of 256:1. If the reference current in the multiplying mode ranges from 16 μ A to 4 mA, the additional error contributions are less than 1.6 μ A. This is well within 8-bit accuracy when referred to full-scale.

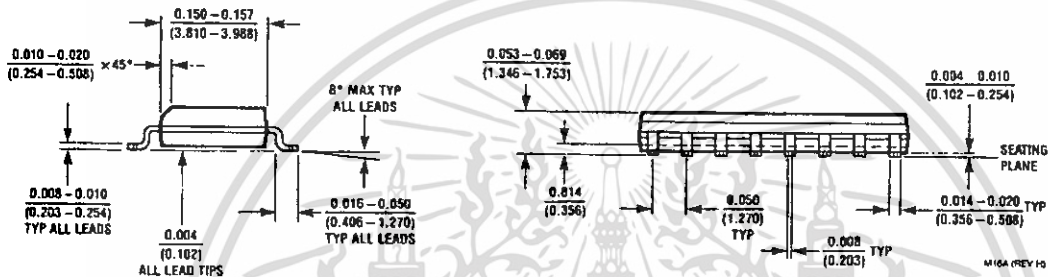
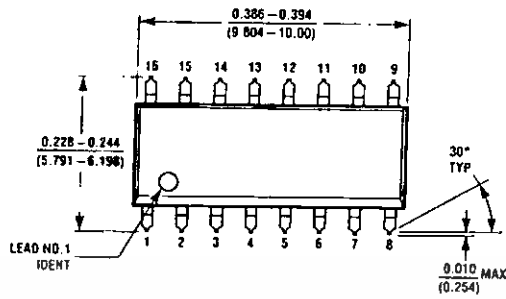
A monotonic converter is one which supplies an increase in current for each increment in the binary word. Typically, the DAC0808 is monotonic for all values of reference current above 0.5 mA. The recommended range for operation with a DC reference current is 0.5 to 4 mA.

SETTLING TIME

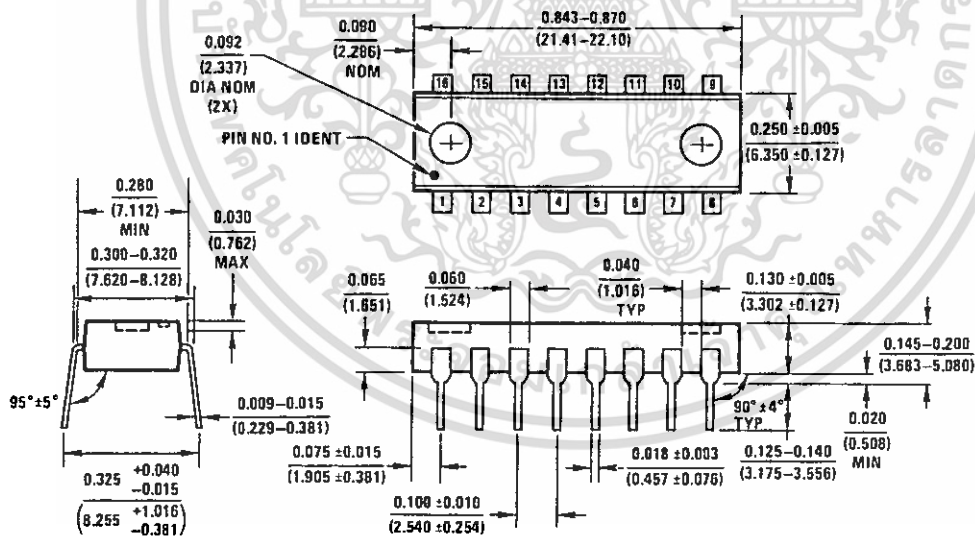
The worst-case switching condition occurs when all bits are switched ON, which corresponds to a low-to-high transition for all bits. This time is typically 150 ns for settling to within $\pm 1/2$ LSB, for 8-bit accuracy, and 100 ns to $1/2$ LSB for 7 and 6-bit accuracy. The turn OFF is typically under 100 ns. These times apply when $R_L \leq 500\Omega$ and $C_O \leq 25$ pF.

Extra care must be taken in board layout since this is usually the dominant factor in satisfactory test results when measuring settling time. Short leads, 100 μ F supply bypassing for low frequencies, and minimum scope lead length are all mandatory.

Physical Dimensions inches (millimeters) unless otherwise noted



Small Outline Package
Order Number DAC0808LCM
NS Package Number M16A



Dual-In-Line Package
Order Number DAC0808
NS Package Number N16A

N16A (REV E)


Notes



LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

 <p>National Semiconductor Corporation Americas Email: support@nsc.com www.national.com</p>	<p>National Semiconductor Europe Fax: +49 (0) 180-530 85 86 Email: europa.support@nsc.com Deutsch Tel: +49 (0) 69 9508 6208 English Tel: +44 (0) 870 24 0 2171 Français Tel: +33 (0) 1 41 91 8790</p>	<p>National Semiconductor Asia Pacific Customer Response Group Tel: 65-2544466 Fax: 65-2504466 Email: ap.support@nsc.com</p>	<p>National Semiconductor Japan Ltd. Tel: 81-3-5639-7560 Fax: 81-3-5639-7507</p>
---	--	---	---

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้